VISIT…

Friedrich Recknagel
Editor

Ecological Informatics

Scope, Techniques and Applications

Second Edition

Springer

Friedrich Recknagel (Ed.)
Ecological Informatics
Scope, Techniques and Applications

Friedrich Recknagel (Ed.)

Ecological Informatics

Scope, Techniques and Applications

2nd Edition

With 174 Figures and a CD-ROM

Springer

This eBook does not include ancillary media that was packaged with the printed version of the book.

EDITOR

ASSOCIATE PROFESSOR FRIEDRICH RECKNAGEL
SCHOOL OF EARTH AND ENVIRONMENTAL SCIENCES
THE UNIVERSITY OF ADELAIDE
5005 AUSTRALIA

E-mail: Friedrich.Recknagel@adelaide.edu.au

ISBN 3-540-43455-0 Springer Berlin Heidelberg New York 1st edition 2003
ISBN 10 3-540-28383-8 **Springer Berlin Heidelberg New York**
ISBN 13 978-3540-28383-6 **Springer Berlin Heidelberg New York**

Library of Congress Control Number: 2005930717

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media
springeronline.com
© Springer-Verlag Berlin Heidelberg 2006
Printed in Germany

The use of general descriptive names, registered names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

Cover design: E. Kirchner, Heidelberg
Production: A. Oelschläger
Typesetting: Camera-ready by the Editor
Printing: Stürtz AG, Gemany
Binding: Stürtz AG, Germany

Printed on acid-free paper 30/2132/AO 5 4 3 2 1 0

To Karina, Melanie, Natalie and Philipp

Preface 2nd Edition

Ecological informatics (ecoinformatics) is an interdisciplinary framework for the processing, archival, analysis and synthesis of ecological data by advanced computational technology (Recknagel 2003). Processing and archival of ecological data aim at facilitating data standardization, retrieval and sharing by means of metadata and object-oriented programming (e.g. Michener *et al.* 1997; Dolk 2000; Sen 2003; Eleveld, Schrimpf and Siegert 2003). Analysis and synthesis of ecological data aim at elucidating principles of information processing, structuring and functioning of ecosystems, and forecasting of ecosystems behaviours by means of bio-inspired computation (e.g. Fielding 1999; Lek and Guegan 2000; Recknagel 2003).

Ecological informatics currently undergoes the process of consolidation as a discipline. It corresponds and partially overlaps with the well-established disciplines bioinformatics and ecological modeling but is taking its distinct shape and scope. In Fig. 1 a comparison is made between ecological informatics and bioinformatics. Even though both are based on the same computational technology their focus is different. Bioinformatics focuses very much on determining gene function and interaction (e.g. Overbeck *et al.* 1999; Wolf et al. 2001), protein structure and function (e.g. Henikoff *et al.* 1999; Lupas, Van Dyke and Stock 1991) as well as phenotype of organisms utilizing DNA microarray, genomic, physiological and metabolic data (e.g. Lockhardt and Winzeler 2000) (Fig. 1a). By contrast ecological informatics focuses to determine population function and interactions as well as ecosystem structure and functioning by utilizing genomic, phenotypic, community, environmental and climate data (e.g. D'Angelo *et al.* 1995; Chon *et al.* 2003; Park *et al.* 2003, Jeong, Recknagel and Joo 2003) (Fig. 1b).

A comparison is made between ecological modeling and ecological informatics in Fig. 2. Even though both rely on similar ecological data they adopt different approaches in utilizing the data. Whilst ecological modeling processes ecological data top down by ad hoc designed statistical or mathematical models (e.g. Straskraba and Gnauck 1985; Jorgensen 1994), ecological informatics infers ecological processes from ecological data patterns bottom up by computational techniques. The cross-sectional area between ecological modeling and ecological informatics reflects a new generation of hybrid models that enable to predict emergent ecosystem structures and behaviours, and ecosystem evolution (e.g. Booth 1997; Downing 1997; Hraber and Milne 1997; Huse, Strand and Giske 1999). Typically those models embody biologically-inspired computation in deterministic ecological models.

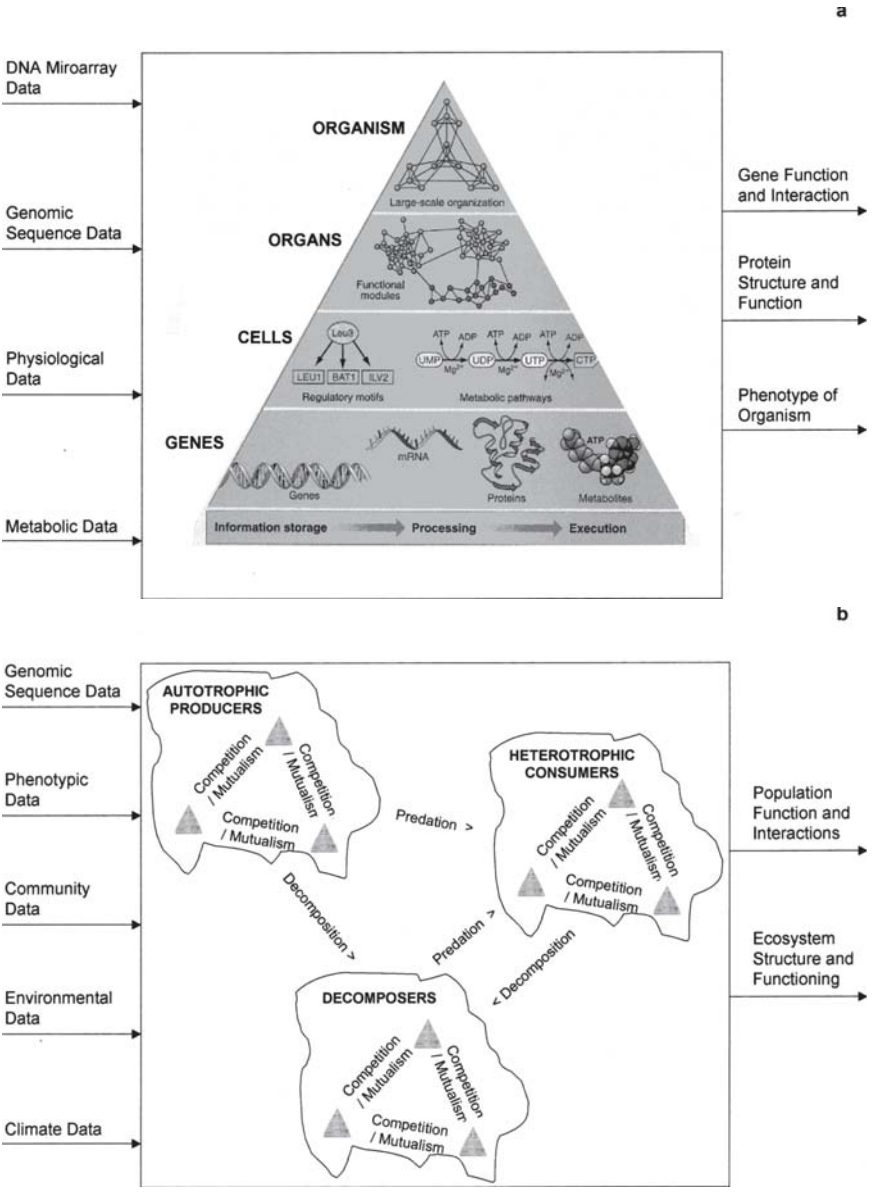


Figure 1. Ecological informatics versus bioinformatics, a) Scope of bioinformatics (modified from Oltvai and Barabasi (2002)), b) Scope of ecoinformatics

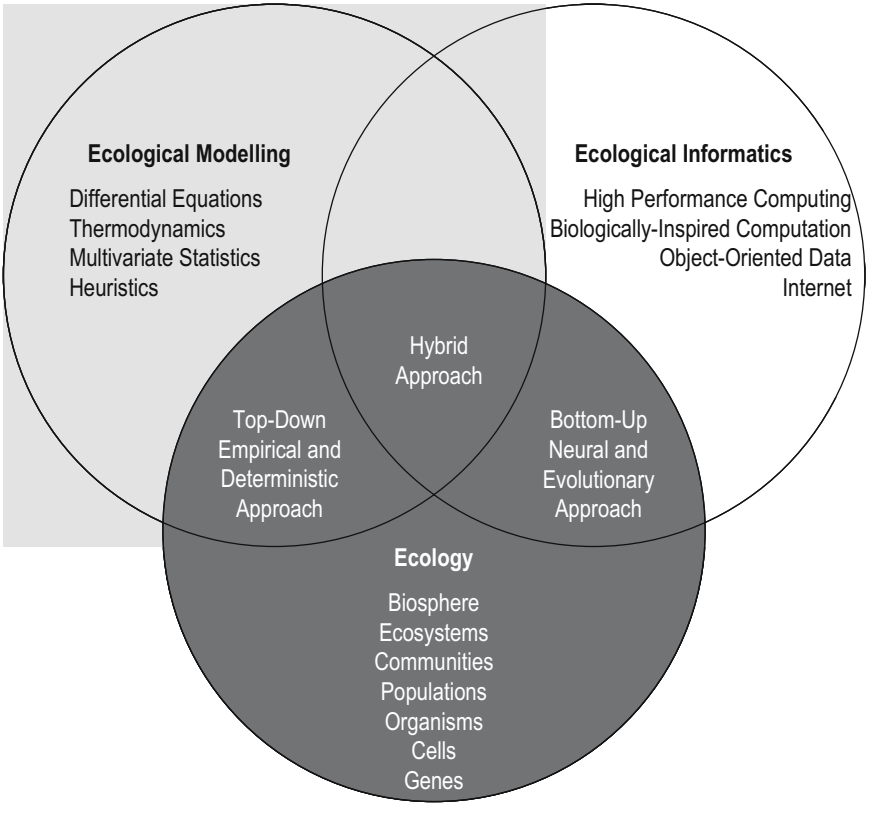


Figure 2. Ecological informatics versus ecological modeling

The term ecological informatics was suggested at the International Conference on Applications of Machine Learning to Ecological Modelling in 2000 (see *Ecological Modelling* 2001, 195) when the International Society for Ecological Informatics ISEI (www.waite.Adelaide.edu.au/ISEI) was founded. Since then an increasing number of researchers and research groups identify with this area, and biennial international conferences are organized by the ISEI. Also the new journal *Ecological Informatics* will be issued by Elsevier in October 2005 (www.elsevier.com/locate/ecolinf).

The contents of the 2nd edition of the book Ecological Informatics has been revised and extended. Two new chapters have been added to Part I: Introduction. Chapter 2 by Bredeweg *et al.* provides an introduction to the novel concept of qualitative reasoning that emerges as an alternative approach to fuzzy logic for automated processing and utilizing of heuristic ecological knowledge. Exemplary applications to population and community dynamics illustrate the potential of the approach. Chapter 7 by Tempesti *et al.* addresses the novel concept of self-

replicating cellular automata inspired by the nature of the genome as the hereditary information of an organism. The authors demonstrate how self-replicating cellular automata can be explored for the design of nano-scale circuits for computer hardware. The paper contributes to the fast growing research on bio-inspired design of both computer software and hardware.

Three new chapters have been added to Part IV: Prediction and Elucidation of Lake and Marine Ecosystems. Chapter 16 by Recknagel *et al.* presents an integrated approach of super- and non-supervised artificial neural networks (ANN) for understanding and forecasting of phytoplankton population dynamics in limnological time series data. The authors complement qualitative ordination and clustering by non-supervised ANN with sensitivity curves from supervised ANN to reveal complex ecological relationships. They apply recurrent supervised ANN for 7-days-ahead forecasting of algal species abundances and succession. Chapter 17 by Cao *et al.* introduces hybrid evolutionary algorithms (HEA) as powerful tools for the discovery of predictive rule sets. The underlying algorithms optimize both the rule structures and multiple parameters. The authors demonstrate that the rule sets discovered in complex limnological time series data achieve not only highly accurate 7-days-ahead forecasting of algal species abundances and succession but provide a high degree of explanation by means of THEN- and ELSE-branch specific sensitivity analysis. A CD with a demo version of HEA is attached and instructions for HEA can be found in the Appendix. Chapter 20 by Atanasova *et al.* demonstrates computational assemblage of ordinary differential equations (ODE) based on an ecological process function library and measured ecological data. The authors document automatically assembled ODE for chlorophyll *a* in a lake and related validation results that indicate possibilities and limitations of the approach.

I want to thank all of the authors who contributed to the book with great enthusiasm and delivered on time. Finally I express my thanks to Dr. Christian Witschel and Agata Oelschlaeger of the Geosciences Editorial Team of the Springer-Verlag for their close collaboration in producing the book

References:

Booth, G., 1997. Gecko: A continuous 2D world for ecological modeling. Artificial Life 3, 147-163.

Chon, T.-S., Park, Y.S., Kwak, I.-S. and E.Y. Cha, 2003. Non-linear approach to grouping, dynamics and organizational informatics of benthic macroinvertebrate communities in streams by artificial neural networks. In: Recknagel, F. (ed.), 2003. Ecological Informatics. Understanding Ecology by Biologically-Inspired Computation. Springer-Verlag, Berlin, Heidelberg, New York, 127-178.

D'Angelo, D.J., Howard, L.M., Meyer, J.L., Gregory, S.V. and L.R. Ashkenas, 1995. Ecological uses of genetic algorithms: predicting fish distributions in complex physical habitats. Can.J.Fish.Aquat.Sci. 52, 1893-1908.

Dolk, D.R., 2000. Integrated model management in the data warehouse area. European Journal of Operational Research 1222, 1999-218.

Downing, K., 1997. EUZONE: Simulating the evolution of aquatic ecosystems. Artificial Life 3, 307-333.

Eleveld, M.A., Schrimpf, W.B.H. and A.G. Siegert, 2003. User requirements and information definition for the virtual coastal and marine data warehouse. Ocean & Coastal Management 46, 487-505.

Fielding, A., 1999. Machine Learning Methods for Ecological Applications. Kluwer, 1-262.

Henikoff, S., Henikoff, J.G. and S. Pietrovski, 1999. Blocks+: a non-redundant database of protein alignment blocks derived from multiple compilations. Bioinformatics 15, 471-479.

Hraber, P. and B.T. Milne, 1997. Community assembly in a model ecosystem. Ecological Modelling 103, 267-285.

Huse, G., Strand, E. and J. Giske, 1999. Implementing behaviour in individual-based models using neural networks and genetic algorithms. Evolutionary Ecology 13, 469-483.

Jeong, K.-S., Recknagel, F. and G.-J. Joo, 2003. Prediction and elucidation of population dynamics of the blue-green algae *Microcystis aeruginosa* and the diatom *Stephanodiscus hantzschii* in the Nakdong River-Reservoir System (South Korea) by a recurrent artificial neural network. In: Recknagel, F. (ed.), 2003. Ecological Informatics. Understanding Ecology by Biologically-Inspired Computation. Springer-Verlag, Berlin, Heidelberg, New York, 195-213.

Jorgensen, S.E., 1995. Fundamentals of Ecological Modelling. Elsevier, Amsterdam, 1-628.

Lek, S. and J-F. Guegan (eds.), 2000. Artificial Neuronal Networks. Application to Ecology and Evolution. Springer, Berlin, Heidelberg, New York, 1-262.

Lockhardt, D. and E. Winzeler, 2000. Genomics, gene expression and DNA arrays. Nature 405, 827-836.

Lupas, A., Van Dyke, M. and J. Stock, 1991. Predicting coiled coils from protein sequences. Science 252, 1162-1164.

Michener, W.K., Brunt, J.W., Helly, J.J., Kirchner, T.B., and S.G.Stanford, 1997. Nongeospatial metadata for the ecological sciences. Ecological Applications 7, 1, 330-342.

Oltavai, Z.N. and A.-L. Barabasi, 2002. Life's complexity pyramid. Science 298, 763-764.

Overbeck , R., Fonstein, M., D'Souza, M., Pusch, G.D. and N. Maltsev, 1999. The use of gene clusters to infer functional coupling. Proc. Natl. Acad. Sci. USA 96, 2896-2901.

Park, Y.-S., Verdonschot, P.F.M., Chon, T.-s., and S. Lek, 2003. Patterning and predicting aquatic macroinvertebrate diversities using artificial neural networks. Water Research 37, 1749-1758.

Recknagel, F. (ed.), 2003. Ecological Informatics. Understanding Ecology by Biologically-Inspired Computation. Springer-Verlag, Berlin, Heidelberg, New York.

Sen, A., 2003. Metadata management: past, present and future. Decision Support Systems 1043, 1-23

Straskraba, M. and A. Gnauck, 1985. Freshwater Ecosystems: Modelling and Simulation. Elsevier, Amsterdam, 1-302.

Wolf, Y.I., Rogozin, I.B., Kondrashov, A.S. and E.V. Koonin, 2001. Genome alignment, evolution of prokaryotic genome organization, and prediction of gene function using genomic context. Genome Research 11, 356-372.

Friedrich Recknagel
Adelaide, 15 May 2005

Preface 1st Edition

In the 50s and 60s cross-sectional data of lake surveys were utilized for steady state assessments of the eutrophication status of lakes by univariate nonlinear regression. This *statistical approach* (see Table 1) became exemplary for river, grassland and forest models and - because of simplicity - widespread for classification of ecosystems.

In the 70s and 80s multivariate time series data were collected from ecosystems such as lakes, rivers, forests and grasslands in order to improve understanding of ecosystem dynamics. Process-based differential equations were used for the computer simulation of food web dynamics and functional group succession. This *differential equation approach* (see Table 1) is still widely used for scenario analysis.

Table 1. Concepts for Ecosystems Analysis, Synthesis and Forecasting

	Statistical Regression Approach	Differential Equations Approach	Computational Approach
Ecosystem Representation	Steady States	Transitional States	Evolving States
Ecosystem Approximation	Univariate Nonlinear / Multivariate Linear	Multivariate Nonlinear	Multivariate Nonlinear
Ecosystem Complexity	Cross-Sectional Nutrient and Abundance Means	Nutrient Cycles and Food Web Dynamics	Species Succession and Ecosystem Evolution
Aquatic Examples	Phosphorus-Chlorophyll Relationship[1,2]; External P-Loading Concept[3]	AQUAMOD[4]; MS-CLEANER[5]; Bierman[6]; Jorgensen[7]; SALMO[8]	Nonlinear Regression[9]; Nonlinear PCA[10]; DELAQUA[11]; ANNA[12]; Evolved Rules[13]; Evolved Equations[14,15]; ECHO[16]; GECKO[17]
Potential Applications	Ecosystem Classification	Scenario Analysis	Ecosystem Forecasting

[1] Sakamoto M (1966) Primary production by phytoplankton community in some Japanese lakes and its dependence on lake depth. Arch. Hydrobiol. 62, 1-28

[2] Dillon P, Rigler F (1974) The phosphorus-chlorophyll relationship in lakes. Limnol.Oceanogr. 19, 135-148

[3] Vollenweider RA (1968) Scientific fundamentals of eutrophication of lakes and flowing waters with special reference to phosphorus and nitrogen. OECD, Paris. OECD/DAS/SCI/68.27

[4] Straskraba M, Gnauck A (1985) Freshwater Ecosystems: Modelling and Simulation. Elsevier, Amsterdam

[5] Park RA, O'Neill RV, Bloomfield JA, Shugart HH, Booth RS, Goldstein RA, Mankin JB, Koonce JF, Scavia D, Adams MS, Clesceri LS, Colon EM, Dettman EH, Hoopes JA, Huff DD, Katz S, Kitchell JF, Koberger RC, La Row EJ, McNaught DC, Petersohn L, Titus JE, Weiler PR, Wilkinson JW, Zahorcak CS (1974) A generalized model for simulating lake ecosystems. Simulation 33-50

[6]Bierman VJ (1976) Mathematical model of the selective enhancement of blue-green algae by nutrient enrichment. In: Canale RP (eds) Modelling Biochemical Processes in Aquatic Ecosystems. Ann Arbour Science Publishers Inc., Ann Arbour, 1-32

[7] Jorgensen SE (1976) A eutrophication model for a lake. Ecol. Modelling 2, 147-162

[8] Recknagel F, Benndorf J (1982) Validation of the ecological simulation model SALMO. Int. Revue Ges.Hydrobiol. 67, 1, 113-125

[9] Lek S, Delacoste M, Baran P, Dimonopoulos I, Lauga J, Aulagnier J (1996) Application of neural networks to modelling nonlinear relationships in ecology. Ecol. Modelling 90, 39-52

[10] Chon TS, Park YS, Moon KH, Cha EY (1996) Patternizing communities by using artificial neural network. Ecol. Modelling 90, 69-78

[11] Recknagel F, Petzoldt T, Jaeke O, Krusche F (1995). Hybrid expert system DELAQUA - a toolkit for water quality control of lakes and reservoirs. Ecol. Modelling 71, 1-3, 17-36

[12] Recknagel F (1997) ANNA - artificial neural network model predicting species abundance and succession of blue-green algae. Hydrobiologia, 349, 47-57

[13] Bobbin J, Recknagel F (2001) Knowledge discovery for prediction and explanation of blue-green algal dynamics in lakes by evolutionary algorithms. Ecol. Modelling 146, 1-3, 253-264

[14] Whigham P, Recknagel F (2001) An inductive approach to ecological time series modelling by evolutionary computation. Ecol. Modelling 146, 1-3, 275-287

[15] Whigham P, Recknagel F (2001) Predicting chlorophyll-a in freshwater lakes by hybridising process-based models and genetic algorithms. Ecol. Modelling 146, 1-3, 243-251

[16] Holland JH (1992) Adaptation in Natural and Artificial Systems. Addison-Wesley, New York

[17] Booth G (1997) Gecko: A continuous 2-D world for ecological modeling. Artif. Life 3, 147-163

Ecosystems analysis, synthesis and forecasting in the past ten years was very much influenced by inventions in computational technology such as high performance computing and biologically-inspired computation. This *computational approach* (see Table 1) allows to discover knowledge in complex multivariate databases for improving both ecosystem theory and decision support.

The present book focuses on the *computational approach* for ecosystems analysis, synthesis and forecasting called *ecological informatics*. It provides the scope and case studies of ecological informatics exemplary for applications of biologically-inspired computation to a variety of areas in ecology.

Ecological Informatics is defined as interdisciplinary framework promoting the use of advanced computational technology for the elucidation of principles of information processing at and between all levels of complexity of ecosystems - from genes to ecological networks -, and the provision of transparent decisions targeting ecological sustainability, biodiversity and global warming.

Distinct features of ecological informatics are: data integration across ecosystem categories and levels of complexity, inference from data pattern to ecological processes, and adaptive simulation and prediction of ecosystems. Biologically-inspired computation techniques such as fuzzy logic, artificial neural networks, evolutionary algorithms and adaptive agents are considered as core concepts of ecological informatics.

Fig. 1 represents the current scope of ecological informatics indicating that ecological data is consecutively refined to ecological information, ecosystem theory and ecosystem decision support by two basic computational operations: data archival, retrieval and visualization, and ecosystem analysis, synthesis and forecasting.

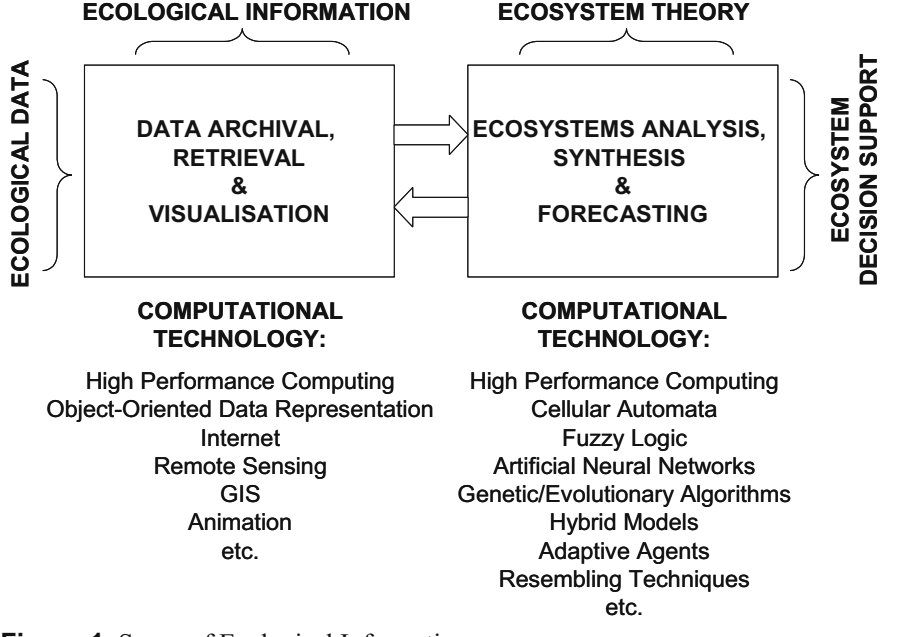


Figure 1. Scope of Ecological Informatics

Computational technologies currently considered being crucial for data archival, retrieval and visualization are:

- High performance computing to provide high-speed data access and processing, and large internal storage (RAM);

- Object-oriented data representation to facilitate data standardization and data integration by the embodiment of metadata and data operations into data structures;
- Internet to facilitate sharing of dynamic, multi-authored data sets, and parallel posting and retrieval of data;
- Remote sensing and GIS to facilitate spatial data visualization and acquisition;
- Animation to facilitate pictorial visualization and simulation.

Following computational technologies are currently considered to be crucial for ecosystems analysis, synthesis and forecasting:
- High performance computing to provide high-speed data access and processing and large internal storage (RAM), and to facilitate high speed simulations;
- Internet and www to facilitate interactive and online simulation as well as software and model sharing;
- Cellular automata to facilitate spatio-temporal and individual-based simulation;
- Fuzzy logic to represent and process uncertain data;
- Artificial neural networks to facilitate multivariate nonlinear regression, ordination and clustering, multivariate time series analysis, image analysis at micro and macro scale;
- Genetic and evolutionary algorithms for the discovery and evolving of multivariate nonlinear rules, functions, differential equations and artificial neural networks; - Hybrid and AI models by the embodiment of evolutionary algorithms in process-based differential equations, the embodiment of fuzzy logic in artificial neural networks or knowledge processing;
- Adaptive agents to facilitate adaptive simulation and prediction of ecosystem composition and evolution.

The present book is an outcome of the *International Conference on Applications of Machine Learning to Ecological Modelling*, 27 November to 1 December 2000, Adelaide, Australia, which concluded with the foundation of the *International Society for Ecological Informatics (ISEI)* (http://www.waite.adelaide.edu.au/ISEI/). The chapters of the present book are based on selected papers of the conference, which are exemplary for current research trends in *ecological informatics*.

Chapters 1 to 5 address principles and ecological application of fuzzy logic, artificial neural networks, genetic algorithms, evolutionary computation and adaptive agents. Salski summarizes concepts of fuzzy logic and discusses applications for knowledge-based modeling, clustering and kriging related to ecotoxicological, geological and population dynamics data. Giraudel and Lek discuss the design and application of unsupervised artificial neural networks for the classification and visualization of multivariate ecological data. They demonstrate the potential of Kohonen-type algorithms by clustering data of forest communities in Wisconsin (USA). Morrall discusses origins and nature of genetic algorithms, and their suitability to induce numerical or rule-based models for ecological applications. Whigham and Fogel provide a scope of evolutionary algorithms and their potential for evolving rules, algebraic and differential equations relevant to ecology. They also address developments on individual and cooperative behaviour, prey-predator algorithms and hierarchical ecosystems

based on evolutionary algorithms. Recknagel reflects on Holland's adaptive agents concept and its potential to more realistically simulate emergent ecosystem structures and behaviours. He distinguishes between individual-based and state variable-based agents, and emphasizes on the embodiment of evolutionary computation in state-variable based agents.

Chapters 6 to 9 provide case studies for the prediction and elucidation of stream ecosystems by means of machine learning techniques. Goethals, Dedecker, Gabriels and de Pauw demonstrate applications of classification trees and artificial neural networks for the bioassessment of the Zwalm river system in Belgium. Schleiter, Obach, Wagner, Werner, Schmidt and Borchardt carried out a comprehensive study of the Breitenbach stream (Germany) based on a variety of unsupervised and supervised learning algorithms for artificial neural networks. They draw interesting conclusions regarding suitability of different algorithms for bioindication of stream habitats and input sensitivity of streams. Chon, Park, Kwak and Cha provide a summary of achievements in the structural classification and dynamic prediction of macroinvertebrate communities in Korean streams by artificial neural networks. They also discuss patterning of organizational aspects of macroinvertebrate communities. Huong, Recknagel, Marshall and Choy study relationships between environmental factors, stream habitat characteristics and the occurrence of macroinvertebrate taxa in the Queensland stream system (Australia) by means of a neural network based sensitivity analysis.

Chapters 10 to 12 contain examples of time series analysis of river water quality by artificial neural networks. Jeong, Recknagel and Joo apply recurrent neural networks to explain and predict the seasonal abundance and succession of different algae species in the River Nakdong (Korea). Validation results reveal a reasonable correspondence between seven days ahead forecasts and observations of algal abundance. Information on favouring conditions and processes for certain algal species discovered by a comprehensive sensitivity analysis comply well with domain knowledge. Bowden, Maier and Dandy combine super- and unsupervised artificial neural networks as well as genetic algorithms for automated input determination of neural networks in order to forecast the abundance of an algae species in the River Murray (Australia). Gevrey, Lek and Oberdorff apply two approaches of sensitivity analysis for the study of riverine fish species by means of artificial neural networks.

Chapters 14 to 17 provide case studies for the application of fuzzy logic, artificial neural networks and evolutionary algorithms to freshwater lakes and marine fishery systems. Karul and Soyupak compare results for the chlorophyll-a estimation in three Turkish lakes achieved by multiple regression and artificial neural networks. Wilson and Recknagel design a generic neural network model for forecasting algal blooms that is validated by means of six lake databases. It considers bootstrapping, bagging and time-lagged training as crucial techniques for minimising prediction errors. Bobbin and Recknagel apply evolutionary algorithms to discover rules for the abundance and succession of blue green algae species in the hypereutrophic Lake Kasumigaura (Japan). Resulting rules correspond with literature findings, reveal hypothetical relationships and are able to predict timing and magnitudes of algal dynamics.

Reick, Gruenewald and Page address the issue of data quality in the context of ecological time-series analysis and prediction. They describe cross-validation and automated training termination of neural networks applied for multivariate time-series predictions of marine zooplankton in the German Northern Sea. Chen combines fuzzy logic and artificial neural networks in order to classify fish stock-recruitment relationships in different environmental regimes near the West Coast Vancouver Island (Canada) and southeast Alaska (USA).

Chapters 18 to 20 provide examples for the classification of ecological images at micro and macro scale by artificial neural networks. Wilkins, Boddy and Dubelaar demonstrate possibilities for the identification of marine microalgae by the analysis of flow cytometric pulse shapes with the help of neural networks. Robertson and Morison applied a probabilistic neural network for the automation of age estimation in three fish species. Thin-sections of sagittal otoliths viewed with transmitted light were used for all species, and the number of opaque increments used to estimate the age. The neural network correctly classified a larger range of age classes. Foody gives a representative summary of neural network algorithms currently used for the pattern recognition and classification of remotely sensed landscape images.

At this point I want to thank all of the authors who responded with great enthusiasm to my request for chapters to the theme of the book and delivered on time. I am also grateful to 24 colleagues and friends in Australia and overseas who significantly improved the quality of chapters by their critical reviews.

Finally I express my thanks to Dr. Christian Witschel and Agata Oelschlaeger of the Geosciences Editorial Team of the Springer Verlag for their close collaboration in producing the book.

Friedrich Recknagel
Adelaide, 15 April 2002

Contents

Part I Introduction 1

1. Ecological Applications of Fuzzy Logic 3
1.1 Fuzzy Sets and Fuzzy Logic 3
1.2 Fuzzy Approach to Ecological Modelling and Data Analysis 4
1.3 Fuzzy Classification: A Fuzzy Clustering Approach 6
1.4 Fuzzy Regionalisation: A Fuzzy Kriging Approach 9
1.5 Fuzzy Knowledge-Based Modelling 9
1.6 Conclusions 12
References 12

2. Ecological Applications of Qualitative Reasoning 15
2.1 Introduction 15
2.2 Why Use QR for Ecology? 16
2.3 What is Qualitative Reasoning? 17
2.3.1 A Working Example 18
2.3.2 World-view: Ontological Distinctions 19
2.3.2.1 Component-based Approach 19
2.3.2.2 Process-based Approach 21
2.3.2.3 Constraint-based Approach 22
2.3.2.4 Suitability of Approaches 23
2.3.3 Inferring Behaviour from Structure 23
2.3.4 Qualitativeness and Representing Time 25
2.3.5 Causality 27
2.3.6 Model-fragments and Compositional Modelling 30
2.4 Tools and Software 30
2.4.1 Workspaces in Homer 31
2.4.2 Building a Population Model 32
2.4.3 Running and Inspecting Models with VisiGarp 35
2.4.4 Adding Migration to the Population model 36
2.5 Examples of QR-based Ecological Modelling 39
2.5.1 Population and Community Dynamics 39
2.5.2 Water Related Models 41
2.5.3 Management and Sustainability 42
2.5.4 Details in Qualitative Algebra 42
2.5.5 Details in Automated Model Building 43

2.5.6 Diagnosis 43
2.6 Conclusion 44
References 44

3. Ecological Applications of Non-Supervised Artificial Neural Networks 49
3.1 Introduction 49
3.2 How to Compute a Self-Organizing Map (SOM) with an Abundance Dataset? 50
3.2.1 A Dataset for Demonstrations 50
3.2.2 The Self-Organizing Map (SOM) Algorithm 52
3.3 How to Use a Self-Organizing Map with an Abundance Dataset? 56
3.3.1 Mapping the Stations 56
3.3.2 Displaying a Variable 58
3.3.3 Displaying an Abiotic Variable 59
3.3.4 Clustering with a SOM 60
3.4 Discussion 63
3.5 Conclusion 65
References 66

4. Ecological Applications of Genetic Algorithms 69
4.1 Introduction 69
4.2 Ecology and Ecological Modelling 70
4.3 Genetic Algorithm Design Details 72
4.4 Applications of Genetic Algorithms to Ecological Modelling 74
4.5 Predicting the Future with Genetic Algorithms 78
4.6 The Next Generation: Hybrids Genetic Algorithms 79
References 80

5. Ecological Applications of Evolutionary Computation 85
5.1 Introduction 85
5.2 Ecological Modelling 86
5.2.1 The Challenges of Ecological Modelling 86
5.2.2 Summary 88
5.3 Evolutionary Computation 88
5.3.1 The Basic Evolutionary Algorithm 90
5.3.2 Summary 93
5.4 Ecological Modelling and Evolutionary Algorithms 93
5.4.1 Equation Discovery 93
5.4.2 Optimisation of Difference Equations 94
5.4.3 Evolving Differential Equations 95
5.4.4 Rule Discovery 95
5.4.5 Modelling Individual and Cooperative Behaviour 97
5.4.6 Predator-Prey Algorithms 100

5.4.7 Modelling Hierarchical Ecosystems 100
5.5 Conclusion 102
References 102

6. Ecological Applications of Adaptive Agents 109
6.1 Introduction 109
6.2 Adaptive Agents Framework 110
6.3 Individual-Based Adaptive Agents 112
6.4 State Variable-Based Adaptive Agents 114
6.4.1 Algal Species Simulation by Adaptive Agents 116
6.4.1.1 Embodiment of Evolutionary Computation in Agents 116
6.4.1.2 Adaptive Agents Bank 117
6.4.2 Pelagic Food Web Simulation by Adaptive Agents 121
6.5 Conclusions 122
Acknowledgements 122
References 123

7. Bio-Inspired Design of Computer Hardware by Self-Replicating Cellular Automata 125
7.1 Introduction 125
7.2 Cellular Automata 126
7.3 Von Neumann's Universal Constructor 128
7.4 Self-Replicating Loops 131
7.5 Self-Replication in the Embryonics Project 132
7.5.1 Embryonics 132
7.5.2 The Tom Thumb Algorithm 136
7.5.2.1 Construction of the Minimal Cell 136
7.5.2.2 Growth and Self-Replication 140
7.5.2.3 The LSL Acronym Design Example 141
7.5.2.4 Universal Construction 144
7.6 Conclusions 145
Acknowledgements 146
References 146

Part II Prediction and Elucidation of Stream Ecosystems 149

8. Development and Application of Predictive River Ecosystem Models Based On Classification Trees and Artificial Neural Networks 151
8.1 Introduction 151
8.2 Study Sites, Data Sources and Modelling Techniques 152
8.2.1 The Zwalm River Basin 152

8.2.2 Data Collection 153
8.2.3 Classification Trees 154
8.2.4 Artificial Neural Networks 155
8.2.5 Model Assessment 156
8.3 Results 157
8.3.1 Classification Trees 157
8.3.1.1 Model Development and Validation 157
8.3.1.2 Application of Predictive Classification Trees for River Management 158
8.3.2 Artificial Neural Networks 160
8.3.2.1 Model Development and Validation 160
8.3.2.2 Application of Predictive Artificial Neural Networks for River Management 162
8.3.2.2.1 Prediction of Environmental Standards 162
8.3.2.2.2 Feasibility Analysis of River Restoration Options 163
8.4 Discussion 164
Acknowledgements 165
References 165

9. Modelling Ecological Interrelations in Running Water Ecosystems with Artificial Neural Networks 169
9.1 Introduction 169
9.2 Materials and Methods 170
9.2.1 Data Base 170
9.2.2 Data Pre-Processing 170
9.2.3 Artificial Neural Network Types 171
9.2.4 Dimension Reduction 171
9.2.5 Quality Measures 171
9.3 Data Exploration with Unsupervised Learning Systems 172
9.4 Correlations and Predictions with Supervised Learning Systems 175
9.4.1 Correlations and Predictions of Environmental Variables 177
9.4.2 Dependencies of Colonisation Patterns of Macro-Invertebrates on Water Quality and Habitat Characteristics 177
9.4.2.1 Aquatic Insects in a Natural Stream, the Breitenbach 177
9.4.2.2 Anthropogenically Altered Streams 180
9.4.3 Bioindication 181
9.5 Assessment of Model Quality and Visualisation Possibilities: Hybrid Networks 182
9.6 Conclusions 183
Acknowledgements 185
References 185

10. Non-linear Approach to Grouping, Dynamics and Organizational Informatics of Benthic Macroinvertebrate Communities in Streams by Artificial Neural Networks 187
10.1 Introduction 187

10.2 Grouping Through Self-Organization 190
10.2.1 Static Grouping 190
10.2.2 Grouping Community Changes 203
10.3 Prediction of Community Changes 207
10.3.1 Multilayer Perceptron with Time Delay 207
10.3.2 Elman Network 211
10.3.3 Fully Connected Recurrent Network 214
10.3.4 Impact of Environmental Factors Trained with the Recurrent Network 218
10.4 Patterning Organizational Aspects of Community 221
10.4.1 Relationships among Hierarchical Levels in Communities 221
10.4.2 Patterning of Exergy 227
10.5 Summary and Conclusions 233
Acknowledgements 234
References 234

11. Elucidation of Hypothetical Relationships between Habitat Conditions and Macroinvertebrate Assemblages in Freshwater Streams by Artificial Neural Networks 239
11.1 Introduction 239
11.2 Study Site 240
11.3 Materials and Methods 240
11.3.1 Data 240
11.3.2 Neural Network Modelling 241
11.3.3 Sensitivity Analysis 242
11.4 Results and Discussion 243
11.4.1 Elucidation of Hypothetical Relationships 243
11.4.2 Discovery of Contradictory Relationships 247
11.4.3 Limitations of the Method 248
11.5 Conclusions 249
References 250

Part III Prediction and Elucidation of River Ecosystems 253

12. Prediction and Elucidation of Population Dynamics of the Blue-green Algae *Microcystis aeruginosa* and the Diatom *Stephanodiscus hantzschii* in the Nakdong River-Reservoir System (South Korea) by a Recurrent Artificial Neural Network 255
12.1 Introduction 255
12.2 Description of the Study Site 256
12.3 Materials and Methods 257
12.3.1 Data Collection and Analysis 257

12.3.2 Modelling the Phytoplankton Dynamics 259
12.3.3 Neural Network Validation and Knowledge Discovery on Algal Succession 261
12.4 Results and Discussion 261
12.4.1 Limnological Aspects and Plankton Dynamics in the Lower Nakdong River 261
12.4.2 Configuring the Neural Network Architecture for Predictability 263
12.4.3 Elucidation of Ecological Hypothesis 265
12.4.3.1 Microcystis aeruginosa 267
12.4.3.2 Stephanodiscus hantzschii 267
12.5 Implications of Ecological Informatics for Limnology 268
12.6 Conclusions 269
Acknowledgements 270
References 270

13. An Evaluation of Methods for the Selection of Inputs for an Artificial Neural Network Based River Model 275
13.1 Introduction 275
13.2 Methods 277
13.2.1 Unsupervised Input Preprocessing 277
13.2.2 Supervised Input Determination 280
13.3 Case Study 282
13.4 Model Development 282
13.4.1 Performance Measures and Model Validation 283
13.4.2 Data Division 283
13.4.3 Determination of Model Inputs 284
13.5 Results and Discussion 284
13.6 Conclusions 290
Acknowledgements 291
References 291

14. Utility of Sensitivity Analysis by Artificial Neural Network Models to Study Patterns of Endemic Fish Species 293
14.1 Introduction 293
14.2 Contribution of Environmental Variables 294
14.3 Application to Ecological Data 295
14.4 Results 296
14.4.1 Predictive Power 296
14.4.2 Sensitivity Analysis 298
14.5 Discussion 302
14.6 Conclusions 304
References 304

Part IV Prediction and Elucidation of Lake and Marine Ecosystems.....307

15. A Comparison between Neural Network Based and Multiple Regression Models in Chlorophyll-a Estimation 309
15.1 Introduction 309
15.1.1 Eutrophication in Water Bodies and Relevant Models 309
15.1.2 Artificial Neural Networks 310
15.1.3 The Use of Artificial Neural Networks in Environmental Modelling..... 311
15.2 Data and Lakes 311
15.3 Methodology..... 313
15.3.1 Artificial Neural Network Approach 314
15.3.1.1 Training Method 314
15.3.1.2 Data Pre-Processing..... 316
15.3.1.3 Improving Generalisation 316
15.3.2 Multiple Regression Modelling Approach..... 317
15.4 Results 317
15.5 Conclusions and Recommendations 320
15.5.1 Conclusions 320
15.5.2 Recommendations..... 321
Acknowledgments..... 322
References..... 322

16. Artificial Neural Network Approach to Unravel and Forecast Algal Population Dynamics of Two Lakes Different in Morphometry and Eutrophication 325
16.1 Introduction 325
16.2 Materials and Methods 326
16.2.1 Study Sites and Data..... 326
16.2.2 Methods 327
16.3 Results 330
16.3.1 Forecasting Seasonal Algal Abundances and Succession..... 330
16.3.2 Relationships between Algal Abundances and Water Quality Conditions..... 331
16.3.3 Relationships between Algal Abundances, Seasons and Water Quality Changes..... 336
16.4 Discussion..... 340
16.4.1 Forecasting Seasonal Algal Abundances and Succession..... 340
16.4.2 Relationships between Algal Abundances, Seasons and Water Quality Changes..... 341
16.5 Conclusions 344
Acknowledgements..... 344
References..... 344

17. Hybrid Evolutionary Algorithm* for Rule Set Discovery in Time-Series Data to Forecast and Explain Algal Population Dynamics in Two Lakes Different in Morphometry and Eutrophication 347
17.1 Introduction 347
17.2 Materials and Methods 348
17.2.1 Study Sites and Data 348
17.2.2 Hybrid Evolutionary Algorithms 349
17.2.2.1 Structure Optimisation of Rule Sets Using GP 351
17.2.2.2 Parameter optimization of Rule Sets Using a General Genetic Algorithm 356
17.2.2.3 Forecasting by Rule Sets 357
17.3 Case Studies Lake Kasumigaura and Lake Soyang 358
17.3.1 Parameter Settings and Measures 358
17.3.2 Results and Discussion 359
17.4 Conclusions 366
References 366

18. Multivariate Time-Series Prediction of Marine Zooplankton by Artificial Neural Networks 369
18.1 Introduction 369
18.2 Generalisation 371
18.3 Automatic Termination of Training 374
18.4 Case Study: Zooplankton Prediction 378
18.5 Conclusions 381
Acknowledgement 382
References 382

19. Classification of Fish Stock-Recruitment Relationships in Different Environmental regimes by Fuzzy Logic Combined with a Bootstrap Re-sampling Approach 385
19.1 Introduction 385
19.2 Fuzzy Stock-Recruitment Model 386
19.2.1 Traditional Stock-Recruitment Model 386
19.2.2 Fuzzy Stock-recruitment Model 388
19.2.2.1 Fuzzy Membership Function (FMF) 389
19.2.2.2 Fuzzy Rules 390
19.2.2.3 Fuzzy Reasoning 391
19.3 Hybrid Optimal Learning and Bootstrap Re-sampling Algorithms 393
19.3.1 Hybrid Optimal Learning Algorithms 394
19.3.2 Bootstrap re-sampling Procedure 396
19.4 Two Real Data Analyses 397
19.4.1 West Coast Vancouver Island Herring Stock 397
19.4.1.1 Data Prescription and Preliminary Analyses 397
19.4.1.2 Fuzzy-SR Model Analysis 398
19.4.1.3 Bootstrap Re-sampling Analysis 400

19.4.2 Southeast Alaska Pink Salmon 402
19.4.2.1 Data Prescription and Preliminary Analysis 402
19.4.2.2 Fuzzy-SR Model Analysis 403
19.4.2.3 Bootstrap Re-sampling Analysis 404
19.5 Summary and Discussion 404
Acknowledgements 406
References 406

20. Computational Assemblage of Ordinary Differential Equations for Chlorophyll-*a* Using a Lake Process Equation Library and Measured Data of Lake Kasumigaura 409
20.1 Introduction 409
20.2 Methods and Materials 410
20.2.1 LAGRAMGE: Computational Assemblage of ODE 410
20.2.2 Domain Knowledge Library for Lake Ecosystems 411
20.2.3 Task Specification 412
20.2.4 Data of Lake Kasumigaura 415
20.2.5 Experimental Framework 416
20.3 Results and Discussion 418
20.3.1 Experiment 1 418
20.3.2 Experiment 2 422
20.3.3 Experiment 3 424
20.4 Conclusions
References 427

Part V Classification of Ecological Images at Micro and Macro Scale 429

21. Identification of Marine Microalgae by Neural Network Analysis of Simple Descriptors of Flow Cytometric Pulse Shapes 431
21.1 Introduction 431
21.2 Materials and Methods 435
21.2.1 Pulse Shape Extraction 435
21.2.2 Data Filtering 435
21.2.3 Data Transformation 435
21.2.4 Principal Component Analysis 436
21.2.5 Neural Network Analysis 438
21.2.6 Hardware and Software 439
21.3 Results 439
21.4 Discussion 441
21.5 Conclusions 441
Acknowledgement 441

References 442

22. Age Estimation of Fish Using a Probabilistic Neural Network 445
22.1 Introduction 445
22.2 Traditional Methods of Age Estimation 445
22.3 Approaches to Automation in Fish Age Estimation 447
22.4 The Application of a Probabilistic Neural Network to Fish Age Estimation 448
22.5 Results 452
22.6 Discussion 454
Acknowledgements 456
References 456

23. Pattern Recognition and Classification of Remotely Sensed Images by Artificial Neural Networks 459
23.1 Introduction 459
23.2 Neural Networks in Remote Sensing 460
23.2.1 Classification Applications 460
23.2.2 Regression Applications 461
23.3 The Neural Networks Used in Remote Sensing 461
23.3.1 Feedforward Neural Networks 462
23.3.1.1 Multi-Layer Perceptron (MLP) 463
23.3.1.2 Radial Basis Function (RBF 464
23.3.1.3 Probabilistic Neural Networks (PNN) 465
23.3.1.4 Generalised Regression Neural Networks (GRNN) 466
23.3.1.5 Other Network Types 467
23.4 Current Status 468
23.4.1 An Example of Neural Networks for Classification 469
23.4.2 Concerns with neural Networks 471
23.5 Conclusions 472
Acknowledgments 473
References 473

Index 479
Appendix 483

Contributors

Nataša Atanasova
Faculty of Civil and Geodetic Engineering, University of Ljubljana, Slovenia
E-mail: natanaso@fgg.uni-lj.si

Lynne Boddy
Cardiff School of Biosciences, Cardiff University, Cardiff CF10 3TL, U.K.
E-mail: BoddyL@Cardiff.ac.uk

Dietrich Borchardt
University of Kassel
Institute of Aquatic Resources Research and Management
Kurt-Wolters-Str. 3, D-34125 Kassel
Germany

Gavin Bowden
Department of Civil and Environmental Engineering,
University of Adelaide, Adelaide 5005
Australia
E-mail: gbowden@civeng.adelaide.edu.au

Bert Bredeweg
University of Amsterdam
Faculty of Science, Informatics Institute
Kruislaan 419
1098 VA Amsterdam
The Netherlands
E-mail: bredeweg@science.uva.nl

Hongqing Cao
School of Earth and Environmental Sciences, University of Adelaide
Adelaide 5005
Australia
E-mail: Hongqing.Cao@adelaide.edu.au

Eui Young Cha
Division of Electronics and Computer Sciences,
Pusan National University, Pusan, Korea

Din Chen
International Pacific Halibut Commission
University of Washington, Seattle WA 98195
USA
E-mail: din@iphc.washington.edu

Tae-Soo Chon
Division of Biological Sciences,
Pusan National University, Pusan
Korea
E-mail: tschon@hyowon.cc.pusan.ac.kr

Satish Choy
Queensland Department of Natural Resources and Mining,
1345 Ipswich Rd, Rocklea 4106
Australia
E-mail: Satish.Choy@dnr.qld.gov.au

Graeme Dandy
Department of Civil and Environmental Engineering
Adelaide University, Adelaide 5005
Australia

A Dedecker
Laboratory of Environmental Toxicology and Aquatic Ecology
Ghent University
J. Plateaustraat 22, B-9000 Ghent
Belgium

N. de Pauw
Laboratory of Environmental Toxicology and Aquatic Ecology
Ghent University
J. Plateaustraat 22, B-9000 Ghent
Belgium

Sašo Džeroski
Jožef Štefan Institute, Ljubljana, Slovenia
E-mail: Saso.Dzeroski@ijs.si

George B.J. Dubelaar
Dubelaar Research Instruments Engineering (DRIE)
Zeelt 2, 2411 DE Bodegraven
The Netherlands

Gary B. Fogel
Natural Selection, Inc., La Jolla, CA 92037
U.S.A.
E-mail: gfogel@natural-selection.com

Giles M. Foody
Department of Geography, University of Southampton
Highfield, Southampton, SO17 1BJ
U.K.
E-mail: G.M.Foody@soton.ac.uk

Wim Gabriels
Laboratory of Environmental Toxicology and Aquatic Ecology
Ghent University
J. Plateaustraat 22, B-9000 Ghent
Belgium
E-mail: wim.gabriels@rug.ac.be

Peter Goethals
Laboratory of Environmental Toxicology and Aquatic Ecology
Ghent University
J. Plateaustraat 22, B-9000 Ghent
Belgium
E-mail: peter.goethals@rug.ac.be

Muriel Gevrey
CESAC (Center d'Ecologie des Systemes Aquatiques et Continentaux)
University of Paul Sabatier, Toulouse
FRANCE
E-mail: gevrey@cict.fr

Jean-Luc Giraudel
CESAC (Center d'Ecologie des Systemes Aquatiques et Continentaux)
University of Paul Sabatier, Toulouse
FRANCE
E-mail: giraudel@montesquieu.u-bordeaux.fr

Antje Gruenewald
University Hamburg, Department of Computer Science
Vogt-Koelln-Str. 30
22527 Hamburg
Germany

Huong Hoang
School of Earth and Environmental Sciences, University of Adelaide
Adelaide 5005
Australia

Kwang-Seuk Jeong
Department of Biology
Pusan National University
Jang-Jeon Dong, Gum-Jeong Gu, Busan, 609-735
Korea
E-mail: pow5150@hotmail.com, pow0606@hananet.net

Gea-Jae Joo
Department of Biology
Pusan National University
Jang-Jeon Dong, Gum-Jeong Gu, Busan, 609-735
Korea
E-mail: gjjoo@pusan.ac.kr

Cueneyt Karul
Department of Environmental Engineering
Middle East Technical University, 06531, Ankara
Turkey

Bomchul Kim
Department of Environmental Science
Kangwon University
Chunchon 200-701
South Korea
E-mail: bkim@kangwon.ac.kr

Boris Kompare
Faculty of Civil and Geodetic Engineering, University of Ljubljana, Slovenia
E-mail: bkompare@fgg.uni-lj.si

Inn-Sil Kwak
Division of Biological Sciences
Pusan National University, Pusan
Korea

Sovan Lek
CESAC (Center d'Ecologie des Systemes Aquatiques et Continentaux)
University of Paul Sabatier, Toulouse
FRANCE
E-mail: lek@cict.fr

Enrico Petraglia
Ecole Polytechnique Federale de Lausanne
School of Computing and Communication Sciences
CH-1015 Lousanne
Switzerland

Holger Maier
Department of Civil and Environmental Engineering
Adelaide University, Adelaide 5005
Australia
E-mail: hmaier@civeng.adelaide.edu.au

Daniel Mange
Ecole Polytechnique Federale de Lausanne
School of Computing and Communication Sciences
CH-1015 Lousanne
Switzerland
E-mail: daniel.mange@epfl.ch

John Marshall
Queensland Department of Natural Resources and Mining
1345 Ipswich Rd., Rocklea 4106
Australia
E-mail: Jonathan.Marshall@dnr.qld.gov.au

Donna Morrall
The Procter & Gamble Co.
Environmental Science Department, Miami Valley Laboratories
P.O. Box 538707, Cincinnati, OH 45253-8707, U.S.A.
E-mail: morrall.dd@pg.com

Alexander K. Morison
Marine and Freshwater Resources Institute
PO Box 114, Queenscliff, VIC 3225
AUSTRALIA
E-mail: Sandy.morison@nre.vic.gov.au

Michael Neumann
University of Giessen
Interdisciplinary Research Centre for Environmental Protection
Heinrich-Buff-Ring 26-32
D-35392 Giessen
Germany
E-mail: mn@mneum.de

Michael Obach
Limnologische Fluss-Station Schlitz der Max-Planck-Gesellschaft
P.O.Box 260, D-36105 Schlitz
Germany
E-mail: michaelo@neuro.informatik.uni-kassel.de

Thierry Oberdorff
Institut dEcologie et de Gestion de la Biodiversite, MNHN,
Lab. Ichtyologie appliquee, 43 rue Cuvier, 75005 Paris
France

Bernd Page
University Hamburg, Department of Computer Science
Vogt-Koelln-Str. 30
22527 Hamburg
Germany

Young Seuk Park
CESAC (Center d'Ecologie des Systemes Aquatiques et Continentaux)
University of Paul Sabatier, Toulouse
FRANCE

Friedrich Recknagel
School of Earth and Environmental Sciences, University of Adelaide
Adelaide 5005
Australia
E-mail: Friedrich.Recknagel@adelaide.edu.au

Christian H. Reick
Alfred-Wegener-Institute
PMB 120161, 27515 Bremerhaven
Germany
E-mail: creick@AWI-Bremerhaven.DE

Simon G. Robertson
Marine and Freshwater Resources Institute
PO Box 114, Queenscliff, VIC 3225
AUSTRALIA
E-mail: simon.robertson@nre.vic.gov.au

Paulo Salles
Universidade de Brasilia
Instituto de Ciências Biológicas
Campus Darcy Ribeiro
Brasilia - DF, 70.910-900, Brasil
E-mail: psalles@unb.br

Ingrid M. Schleiter
University of Kassel
Institute of Aquatic Resources Research and Management
Kurt-Wolters-Str. 3, D-34125 Kassel
Germany
E-mail: schleiter@hrz.uni-kassel.de

Hans-Heinrich Schmidt
Limnologische Fluss-Station Schlitz der Max-Planck-Gesellschaft
P.O.Box 260, D-36105 Schlitz
Germany

Selcuk Soyupak
Department of Environmental Engineering
Middle East Technical University
06531 Ankara
Turkey
E-mail: soyupak@metu.edu.tr

Andre Stauffer
Ecole Polytechnique Federale de Lausanne
School of Computing and Communication Sciences
CH-1015 Lousanne
Switzerland

Noriko Takamura
National Institute for Environmental Sciences
Tsukuba 305-0053
Japan
E-mail: noriko-t@nies.go.jp

Gianluca Tempesti
Ecole Polytechnique Federale de Lausanne
School of Computing and Communication Sciences
CH-1015 Lousanne
Switzerland
E-mail: Gianluca.Tempesti@epfl.ch

Ljupčo Todorovski
Jožef Štefan Institute, Ljubljana, Slovenia
E-mail: Ljupco.Todorovski@ijs.si

Ruediger Wagner
Limnologische Fluss-Station Schlitz der Max-Planck-Gesellschaft
P.O.Box 260, D-36105 Schlitz, Germany
E-mail: RWAGNER@MPIL-SCHLITZ.MPG.DE

Heinrich Werner
University of Kassel,
Dept. of Mathematics and Computer Sciences
Research Group Neural Networks
Heinrich-Plett-Str. 40, D-34132 Kassel
Germany
E-mail: werner@neuro.informatik.uni-kassel.de

Malcolm F. Wilkins
Cardiff School of Biosciences, Cardiff University
Cardiff CF10 3TL
U.K.
E-mail: WilkinsMF@Cardiff.ac.uk

Amber Welk
School of Earth and Environmental Sciences, University of Adelaide
Adelaide 5005
Australia
E-mail: Amber.Welk@adelaide.edu.au

Peter Whigham
University of Otago, Department of Information Science
Dunedin
New Zealand
E-mail: pwhigham@infoscience.otago.ac.nz

Part I

Introduction

Chapter 1

Ecological Applications of Fuzzy Logic

A. Salski

1.1 Fuzzy Sets and Fuzzy Logic

The Fuzzy Set Theory developed by L. Zadeh (Zadeh 1965) as a possible way to handle uncertainty is particularly useful for the representation of vague expert knowledge and processing uncertain or imprecise information. The Fuzzy Set Theory is based on an extension of the classical meaning of the term "set" and formulates specific logical and arithmetical operations for processing information defined in the form of fuzzy sets and fuzzy rules.

The theory of fuzzy sets deals with subsets of a given universe, where the transition between full membership and no membership is gradual. Therefore the boundaries of fuzzy sets are not sharp. An example of a fuzzy set is the set A of all large carps as a subset of all carps in Lake Belau (Salski and Kandzia 1996). Traditionally, the grade of membership 1 is assigned to those objects of the universe that fully belong to a set, while 0 is assigned to objects that do not belong to the set. In traditional set theory, the sets considered are defined as collections of objects having some property, for example the property "carp in Lake Belau". The property "large carp in Lake Belau" does not constitute a set in the usual sense, the property does not offer a precisely defined criterion of membership. Intuitively, a fuzzy set is a collection of objects that admits the possibility of partial membership in it. Thus a fuzzy set A in a given universe is characterized by a function $\mu_A(x)$ termed "the grade of membership of x in A". We shall assume that the values of $\mu_A(x)$ are elements of the interval [0,1] , with the grades 1 and 0 representing full membership and non-membership, respectively. $\mu_A(x)$ is called the membership function of A.

Fuzzy logic is based on the extension of the rules of conventional logic. This extension enables us to process fuzzy rules in the “IF – THEN” form with fuzzy sets in the premise and conclusion parts of these rules. These fuzzy sets represent imprecise expressions used by experts to describe their knowledge. Therefore fuzzy inference methods are particularly useful to work with such a vague knowledge representation. The main difference to conventional methods is that the Fuzzy Set Theory offers inference methods for the calculation of the conclusion values of rules when the premises of these rules are not completely fulfilled.

There are a lot of good books containing details about fuzzy sets and fuzzy logic such as Zimmermann (1993), Kruse et al. (1995), Bárdossy and Duckstein (1995) and Pedrycz (1996).

1.2 Fuzzy Approach to Ecological Modelling and Data Analysis

Heterogeneity and uncertainty belong to the characteristic properties of the data stored in ecological data bases and ecological information systems. Ecologists collect and use information from various heterogeneous data and knowledge sources - sources of objective (mostly quantitative) information, e.g. measurement and calculation, and sources of subjective (often only qualitative) information, e.g. expert knowledge and subjective evaluations instead of measurement data. Therefore in many fields of ecological research ecologists have to work with a necessarily subjective mixture of quantitative and qualitative information. Not all ecological parameters are measurable (for example the number of fish in a particular lake); the values of such parameters can be obtained by special estimation or evaluation methods, which are often of a subjective character. Ecological data can also have different structures and formats (e.g. time series and spatial data).

The problem of uncertainty often appears in ecological modelling, in particular it concerns the uncertainty of data and vaguely defined expert knowledge. A large inherent uncertainty of ecological data results from the presence of random variables, incomplete or inaccurate data, approximate estimations instead of measurements (due to technical or financial problems) or incomparability of data (resulting from varying measurement or observation conditions). There are a number of ways to deal with uncertainty problems, e.g. probabilistic inference networks (Pearl 1988) or belief intervals (Shafer et al. 1990). One of the most successful methods of dealing with uncertainty is the fuzzy approach. Fuzzy approach does not mean a particular method but the integration of a fuzzy concept into conventional methods of knowledge processing and data analysis. That means an extension of conventional methods, which is capable of utilising imprecise, heterogeneous and uncertain data. Compared to conventional methods the fuzzy approach enables us to make better use of imprecise ecological data and vague expert knowledge in two ways:

- the representation and handling of imprecise data defined as fuzzy sets,
- the representation and processing of vague knowledge in the form of linguistic rules with imprecise terms defined as fuzzy sets.

Ecological data or classes of ecological objects can be defined as fuzzy sets with no sharply defined boundaries, which reflects better the continuous character of nature. Fuzzy sets can be used to handle uncertainty of data and fuzzy logic to handle inexact reasoning. Fuzzy logic allows working with uncertain knowledge about relations between ecosystem components and building models based on this

type of information.

Ecological modelling and data analysis are the main application areas of the fuzzy set theory in ecological research. The integration of the fuzzy inference mechanisms and the expert system technique provides development tools for fuzzy expert systems and fuzzy knowledge-based models of ecological processes (Salski 1999). The evolution of conventional knowledge-based systems into fuzzy systems (adding imprecision or uncertainty handling to conventional systems) makes the extension of their application area for complex ecological problems possible (Kampichler et al. 2000; Freyer 2000; Zhu et al. 1996; Bock and Salski 1996). There are also other fuzzy approaches to ecological modelling, e.g. the fuzzy statistical approach to ecological assessments (Li 2001), the fuzzy differential equations for fuzzy modelling in population dynamics (Barros et al. 2000) or ecological impact analysis using fuzzy logic (Enea et al. 2001; Silvert 1997). The fuzzy memberships can be also used as environmental indices (Silvert 2000) or as a fuzzy association degree in the ecosystem modelling (Liu 2001). There are also an increasing number of other combined approaches, which result from linking the fuzzy approach with other techniques, e.g.:

- fuzzy approach with neural networks for assessment in spatial decision making (Zheng 2001) or for habitat modelling in agricultural landscapes (Wieland et al. 1996),
- fuzzy modelling with conventional dynamic programming to optimal biological control of a greenhouse mite (Cheng et al. 1996),
- fuzzy approach with linear programming for the optimization of land use scenarios (Salski et al. 2001),
- fuzzy approach with probabilistic uncertainty to model climate-plant-herbivore interactions in grassland ecosystems (Wu et al. 1996),
- fuzzy approach with three-dimensional modelling technique (Ameskamp 1997).

The next important research field is handling uncertainty in geographic information systems, that means dealing with fuzziness in reasoning with spatial data (Dragicevic 2000; Guesgen 2000) and in the assignment of locations to classes (Burrough 2000; MacMillan 2000) or fuzziness in the definitions of object boundaries (Cross 2000).

Some application examples of a fuzzy approach to ecological modelling and data analysis are presented in this paper, namely fuzzy clustering as a tool for fuzzy classification of ecological data, fuzzy kriging as a method of fuzzy interpolation of spatial data and fuzzy knowledge-based modelling.

Fuzzy classification and fuzzy geostatistik belong to the main problems of the analysis of ecological data. Conventional classification methods based on Boolean logic ignore the continuous nature of ecological parameters and the uncertainty of data, which can result in misclassification. Fuzzy classification, which means the division of objects into classes that do not have sharply defined boundaries, can be carried out in various ways, for example:

- application of fuzzy arithmetical and logical operations, e.g. to determine land suitability (Burrough et al. 1992),

- fuzzy clustering, e.g. to classify some crop growth parameters (Marsili-Libelli 1994) or to classify existing chemicals according to their ecotoxicological properties (Friederichs et al. 1996).

Compared to conventional classification methods fuzzy clustering methods enable a better interpretation of the data structure.

Spatial data is an essential part of ecological data. The fuzzy extension of the interpolation procedure for spatial data, the so-called fuzzy kriging, can be mentioned as an example of fuzzy approach to spatial data analysis (Bárdossy 1989; Diamond 1989; Piotrowski et al. 1996). Fuzzy kriging is a modification of the conventional kriging procedure; it utilizes exact (crisp) measurement data as well as imprecise estimates obtained from an expert and defined as fuzzy numbers. Regionalization of ecological parameters based on fuzzy kriging reflects better the imprecision of input data.

Fuzzy knowledge-based modelling can be particularly useful where there is no analytical model of the relations to be examined or where there is an insufficient amount of data for statistical analysis, or where the degree of uncertainty of these data is very high (Salski 1992; Salski et al. 1996; Li 1996; Daunicht et al. 1996; Bárdossy and Duckstein 1995; Pedrycz 1996; Bock and Salski 1998). In these cases the only basis for modelling is the expert knowledge, which is often uncertain and imprecise.

1.3 Fuzzy Classification: A Fuzzy Clustering Approach

Conventional clustering methods definitely place an object within only one cluster. With fuzzy clustering this is no longer essential, since the membership value of this object can be split up between different clusters. In comparison to conventional clustering methods the distribution of the membership values provides additional information - the membership values of a particular object can be interpreted as the degree of similarity between this object and the respective clusters (Salski and Kandzia 1996).

Classifying existing chemicals according to their ecotoxicological properties (Friederichs et al. 1996) can be taken as an application example of the fuzzy cluster analysis. The large number of existing chemicals makes it necessary to select representative chemicals which reflect the relevant properties of possibly a major group of compounds. Therefore the main tasks of this application are:

- to find distinguishable clusters with characteristic properties,
- to find chemicals representative for each cluster,
- to examine the role of different parameters for clustering.

Compared to conventional clustering methods the fuzzy clustering technique is more appropriate to handle the uncertainty of ecotoxicological data, which results, for example, from the difficult comparability of these data. The analysis of the partition efficiency indicators was used to choose the fuzzifier value and the determination of the optimal number of clusters, e.g.:

- partition entropy (should be minimal),
- partition coefficient, where values closer to 1 indicate the "better" partition,
- non-fuzziness index, indicating the “best” partition by the highest value, independently of the number of clusters.

The normalized values of these indicators for cluster numbers between 4 and 8 and fuzzifier values of 1.3 and 1.6 are presented in Figure 1.1. Five clusters can be taken as the "optimal" number of clusters for a fuzzifier of 1.3 - whereas a fuzzifier of 1.6 does not lead to a clear statement.

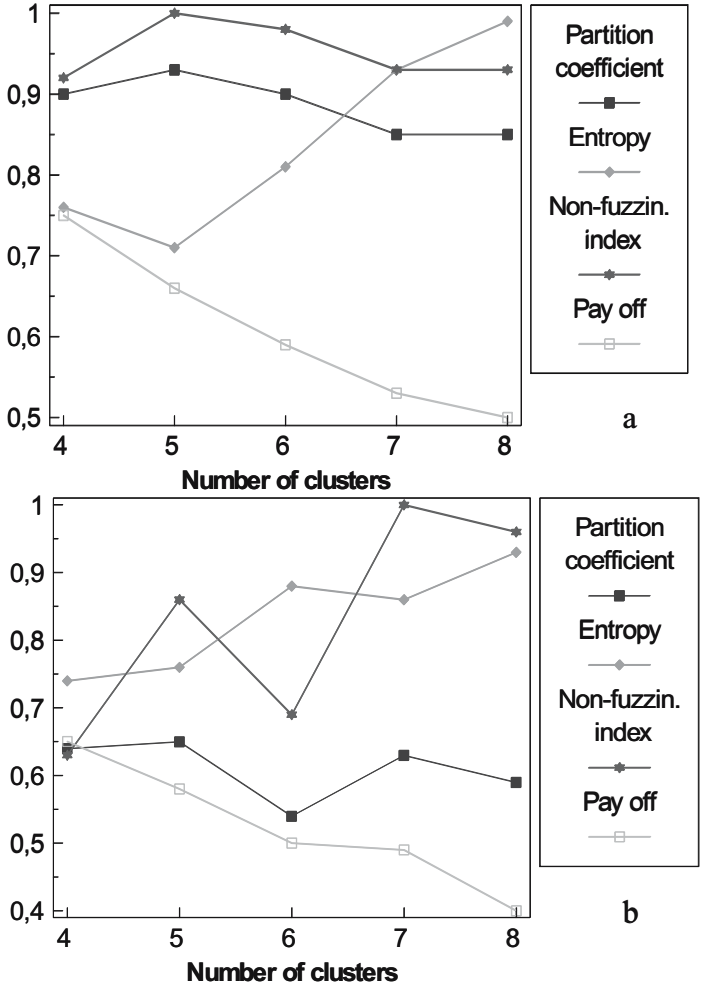


Fig. 1.1. Partition efficiency indicators for fuzzifier values of 1.3 (left) and 1.6 (right) (Friederichs et al. 1996).

The fuzzy partition of 24 chemicals (as a part of a set with more than 200 chemicals) in 5 clusters is presented in the Table 1.1. The numbers in boldface show the highest membership

Table 1.1. Final clustering partition of the 24 chemicals (Friederichs et al. 1996). (The numbers in bold-face show the highest membership values; the membership values with a membership 0.10 to different clusters are underlined).

Cluster 1	**1**	**2**	**3**	**4**	**5**		
<15>Diphenylamine----------			**0.98**	0.00	0.00	0.01	0.01
<27>o-Dianisidine----------			**0.96**	0.00	0.00	0.03	0.01
<30>3,3Dichlorobenzidine----------			**0.69**	0.00	0.08	0.12	0.11
<55>Chlorotoluidine----------			**0.98**	0.00	0.01	0.01	0.00
<74>2-Mercaptobenzothiazole------			**0.99**	0.00	0.00	0.01	0.00
Cluster 2			**1**	**2**	**3**	**4**	**5**
<72>Trichlormethylbenzene--------			0.00	**1.00**	0.00	0.00	0.00
<73>Benzoylchloride----------				0.00	**0.99**	0.00	0.01	0.00
Cluster 3			**1**	**2**	**3**	**4**	**5**
<67>Diethylenglykoldimethylether			0.01	0.00	**0.98**	0.01	0.00
<68>Haxanedioic acid----------			0.04	0.00	**0.60**	0.16	0.19
<70>Acetic acid anhydride----------			0.05	0.21	**0.52**	0.11	0.11
<84>N,N-Dimethylformamide------			0.00	0.00	**0.99**	0.01	0.00
Cluster 4			**1**	**2**	**3**	**4**	**5**
<01>Chloroform----------			0.00	0.00	0.00	**0.96**	0.04
<03>Pentachlorophenol----------			0.04	0.00	0.00	**0.68**	0.28
<10>p-Nitromethoxybenzene------			0.00	0.00	0.00	**0.99**	0.01
<20>Tris-(2-chloroethyl)phosphate			0.11	0.00	0.04	**0.73**	0.12
<24>Benzene----------			0.03	0.00	0.04	**0.72**	0.21
<59>Nitrobenzene----------			0.00	0.00	0.00	**0.99**	0.01
Cluster 5			**1**	**2**	**3**	**4**	**5**
<08>Dichlorobenzene----------			0.00	0.00	0.00	0.02	**0.98**
<13>Nonylphenol----------			0.00	0.00	0.00	0.01	**0.99**
<17>1,2,4-Trichlorobenzene------			0.00	0.00	0.00	0.07	**0.93**
<18>Ditolyl ether----------			0.01	0.00	0.01	0.04	**0.94**
<23>Tributylamine----------			0.07	0.00	0.01	0.36	**0.56**
<25>Hexachloropentadiene--------			0.09	0.01	0.01	0.25	**0.64**
<75>2-Nitrophenol----------			0.02	0.00	0.03	0.17	**0.78**

values (membership values with a membership ≥ 0.10 to different clusters are underlined). The analysis of these results permits to recognize chemicals which may serve as representatives for a particular cluster (names in bold-face) and the characteristic properties of these clusters. For example “diphenylamine” can be taken as a representative for cluster 1, as its membership to the cluster 1 is close to 1. The description of the properties which are characteristic for a particular cluster can be found in (Friederichs et al. 1996).

1.4
Fuzzy Regionalization: A Fuzzy Kriging Approach

Kriging belongs to the most popular methods of spatial interpolation, but its application is often restricted owing to an insufficient amount of data. If the number of available measurements is too low for conventional kriging methods, the data set can be supplemented using additional imprecise data subjectively estimated by an expert. Fuzzy kriging utilizes exact (crisp) measurement data as well as imprecise estimates obtained from an expert (Bárdossy et al. 1989; 1990; Diamond 1989; Kacewicz 1994). The imprecision and uncertainty of these estimates can be handled with the fuzzy sets. The logical structure of this fuzzy kriging procedure with both crisp and fuzzy data and a crisp theoretical variogram is shown in Figure 1.2. The zigzag line marks stages with fuzzy data input in form of fuzzy numbers. At two stages fuzziness is introduced into the calculation. First, fuzziness in the input values causes fuzziness in the experimental variogram. An expert takes the experimental variogram and its fuzziness into account when fitting the crisp theoretical variogram. Second, the fuzzy input values are used at the final step of kriging, namely at the calculation of the interpolated values. Therefore, if the input data set contains at least one fuzzy number the kriging results have the form of fuzzy numbers, too.

As an example of an application of fuzzy kriging in spatial interpolation of geological data a fuzzy kriging interpolation of hydraulic-conductivity values from an aquifer in northwestern Germany could be mentioned (Piotrowski et al. 1996). Because of a high spatial variability of data and irregular distribution of data points the modification of the original data set was necessary. After supplementing the original data set (557 boreholes) with 30 imprecise (fuzzy) points the kriging variance has been significantly reduced. The Fuzzy Evaluation and Kriging System FUZZEKS developed at the University in Kiel (Bartels 1997) has been used as a tool for spatial interpolation. The authors of this application consider the fuzzy kriging approach in interpolation hydrogeological parameters as an important tool with a potential of quantifying vast pieces of information available as expert knowledge.

1.5
Fuzzy Knowledge-Based Modelling

As mentioned above, fuzzy knowledge-based modelling can be particularly useful in cases where the relations between the components of an ecosystem are not exactly known or where we do not have any analytical model for these relations, or where we have an insufficient amount of data for statistical analysis. Ecologists often use vague and ill-defined natural language to describe their knowledge (Salski 1992; 1999). Therefore this knowledge can be represented by a set of linguistic "IF -THEN" rules, which can be interpreted as a linguistic description of

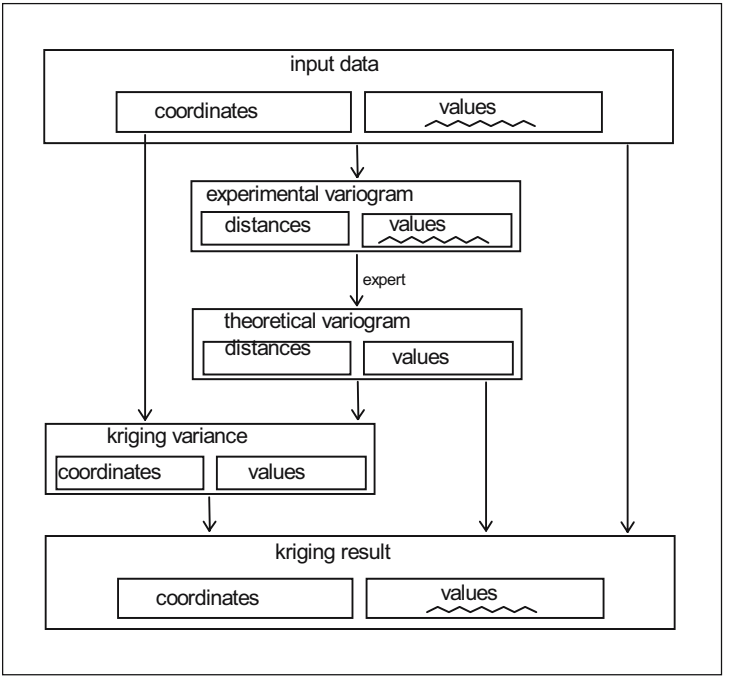


Fig. 1.2. Logical structure of fuzzy kriging with both crisp and fuzzy data (zigzag lines indicate fuzziness of data; Bartels 1997).

the relation between the input and output of a model. It can be used as a basis for the calculation of the output values of the model.

As an application example a fuzzy knowledge-based model of population dynamics of the Yellow-necked mouse (*Apodemus flavicollis*) in a beech forest can be mentioned (Bock and Salski 1998). Animal weight, food availability and soil surface moisture are the most important factors affecting the population dynamics of the Yellow-necked mouse in a beech forest. The relationships between these factors and the population dynamics of these small mammals are not exactly known. Due to technical problems associated with collecting data for a free ranging animal population there was a high degree of uncertainty with part of the available data. That was the reason for employing unconventional modelling methods based on the linguistic description of the process dynamics.

Figure 1.3 shows the structure of this fuzzy dynamic model with the state variable "abundance". The prediction of abundance at time (k+1) is based on the values of abundance, food availability, soil surface moisture and animal weight at time (k). The initial value A_o of the state variable and the initial values of the input variables "food", "moisture" and "weight" (F_o, M_o and W_o, respectively) have to be provided. Then we can calculate the values of "abundance" in successive moments in time (k = 1,2 ,3,....) for given values of the input variables. Each

prediction step (difference between moments in time) represents a period of two months.

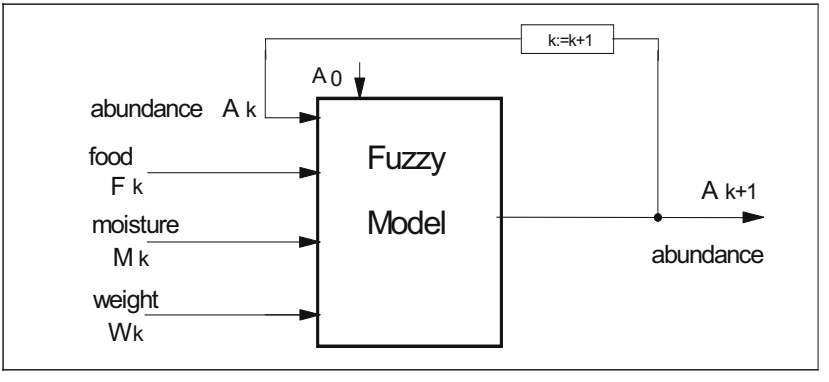


Fig. 1.3. The structure of a fuzzy knowledge-based model of the population dynamics of the Yellow-necked mouse (*Apodemus flavicollis*) in a beech forest (Bock and Salski 1998)

The state variable "abundance" and the variables "weight" and "food" are defined as linguistic variables. Seven fuzzy sets were determined for the variable "abundance" and three fuzzy sets for the input variables "food" and "weight". The input variable "moisture" is defined as a symbolic variable, which means that its values can only be symbolic statements (like dry, average and wet).

The knowledge base of this model contains about 100 linguistic rules in the "IF-THEN" form, for example:

IF the current value for "abundance" is "low"
AND "weight" is "average"
AND "food" is "high"
AND "moisture" is "average"
THEN "abundance" in the next prediction step is "high".

The linguistic terms "low", "high", etc. in the premise and conclusion parts of the rules are determined as fuzzy sets. The definition of fuzzy sets and the formulation of linguistic rules are of a subjective character. The knowledge base of the model has been created using the Modelling Support System FLECO (Salski and Kandzia 1996). The simulation results were calculated for imprecise input values (e.g. "high") of the variable "food" and compared to the field study results. It was difficult to estimate the values of the variable “food” more precisely, however a fuzzy logic approach enables us to make use of such imprecise information. The difference between simulation and field study results is no bigger than 10-15%. The detailed simulation results and a model description can be found in (Bock and Salski 1998).

Another application of this approach can be found in Recknagel et al (1994) where fuzzy rule sets were used for the forecasting of monthly occurrence of algal functional groups in freshwater lakes.

1.6 Conclusions

Heterogeneous and imprecise ecological data and vague expert knowledge can be integrated more effectively using fuzzy approach. Fuzzy logic provides the means to combine numerical data and linguistic statements and to process both of them in one simulation step. Fuzzy sets with no sharply defined boundaries reflect better the continuous character of nature. The number of applications of fuzzy sets and fuzzy logic in ecological modelling and data analysis is constantly growing.

There also are an increasing number of applications of hybrid systems which combine the fuzzy techniques with other techniques, e.g. probabilistic approach, linear programming, neural networks, cellular automata or GIS technique. An increasing interest in the development of fuzzy expert systems for environmental management and engineering can also be expected.

References

Ameskamp M (1997) Three-dimensional rule-based continuous soil modelling. Ph.D. Thesis, Institut für Informatik und Praktische Mathematik, Christian- Albrechts-Universität, Kiel

Bárdossy A, Bogardi I, Kelly WE (1989) Geostatistics utilizing imprecise (fuzzy) information. Fuzzy Sets and Systems, 31/3: 311-327

Bárdossy A, Bogardi I, Kelly WE (1990) Kriging with imprecise (fuzzy) variograms, I: theory. Math. Geology, 22/3: 63-79

Bárdossy A, Duckstein L (1995) Fuzzy rule-based modeling with applications to geophysical, biological and engineering systems. CRC Press, Boca Raton

Barros LC, Bassanezi RC, Tonelli PA (2000) Fuzzy modelling in population dynamics. Ecological Modelling, 128: 27-33

Bartels F (1997) Ein Fuzzy-Auswertungs- und Krigingsystem für raumbezogene Daten. Diplomarbeit, Inst. für Informatik und Praktische Mathematik, Universität Kiel

Bezdek JC (1980) A convergence theorem for the fuzzy c-means clustering algorithms. IEEE Trans. PAMI, PAMI-2(1): 1-8

Bock W, Salski A (1998) A fuzzy knowledge-based model of population dynamics of the Yellow-necked mouse (*Apodemus flavicollis*) in a beech forest. Ecological Modelling, 108:155-161

Burrough PA, Macmillan RA, Van Deursen (1992) Fuzzy classification methods for determination land suitability from soil profile observations and topography. Journal of Soil Science, 43: 193-210

Burrough PA, van Gaans PFM, MacMillian RA (2000) High-resolution landform classification using fuzzy k-means. Fuzzy Sets and Systems, 113: 37-52

Cheng Z, Horn DJ, Lindquist RK, Taylor RAJ (1996) Automated soil inference under fuzzy logic, Ecological Modelling, 90/2: 111-122

Cross V, Firat A (2000) Fuzzy objects for geographical information systems. Fuzzy Sets and Systems, 113: 19-36

Daunicht W, Salski A, Nöhr P, Neubert C (1996) A fuzzy knowledge-based model of the annual production of Skylarks. Ecological Modelling, 85: 67-74

Diamond P (1989) Fuzzy kriging. Fuzzy Sets and Systems, 33/3: 315-332

Dragicevic S, Marceau D J (2000) An application of fuzzy logic reasoning for GIS temporal modelling of dynamic processes. Fuzzy Sets and Systems, 113: 69-80

Enea M, Salemi G (2001) Fuzzy approach to the environmental impact evaluation. Ecological Modelling, 135: 131-147

Freyer B, Reisner Y, Zuberbühler D (2000) Potential impact model to assess agricultural pressure to landscape ecological functions. Ecological Modelling, 130:121-129

Friderichs M, Fränzle O, Salski A (1996) Fuzzy clustering of existing chemicals according to their ecotoxicological properties. Ecological Modelling, 85/1: 27-40

Guesgen HW, Albrecht J (2000) Imprecise reasoning in geographic information systems. Fuzzy Sets and Systems, 113: 121-131

Kampichler Ch, Barthel J, Wieland R (2000) Species density of foliage-dwelling spiders in field margins: a simple, fuzzy rule-based model. Ecological Modelling, 129: 87-99

Li B-L (ed.) (1996) Fuzzy Modelling in Ecology. Ecological Modelling, special issue, 90/2

Li B-L (2001) Fuzzy statistical and modelling approach to ecological assessments. In: M. E. Jensen and P.S. Bourgeron (eds), A guidebook for integrated ecological assessments, Springer, New York: 211-220

Liu W-Y, Song N (2001) The fuzzy association degree in semantic data models. Ecological Modelling, 117: 203-208

Kacewicz M (1994) “Fuzzy" geostatistics - an intergration of qualitative discription into spatial analysis. In: Dimitrakopoulos, R., (Editor), Geostatistics for the next century, Kluwer Academic Publishers, Dordrecht: 448-463

Kruse R, Gebhardt JG, Klawonn F (1995) Fuzzy-Systeme. Teubner, Stuttgart

Marsili-Libelli S (1994) Fuzzy clustering of ecological data. Coenoses, 4/2: 95-106

MacMillian RA, Pettapiece WW, Nolan SC, Goddard TW (2000) A generic procedure for automatically segmenting landforms into landform elements using DEMs, heuristic rules and fuzzy logic. Fuzzy Sets and Systems, 113: 81-109

Pearl J (1988) Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann, San Mateo, CA

Pedrycz W (1996) Fuzzy modelling, Paradigms and practice. Kluwer Academic Publishers, Boston

Piotrowski JA, Bartels F, Salski A, Schmidt G (1996) Geostatistical regionalization of glacial aquitard thicknessin northwestern Germany, based on fuzzy kriging., Mathematical Geolog,y, 28/4: 437-452

Recknagel F, Petzoldt T, Jaeke O, Krusche F (1995). Hybrid expert system DELAQUA - a toolkit for water quality control of lakes and reservoirs. Ecol. Modelling 71, 1-3, 17-36

Salski A (1992) Fuzzy knowledge-based models in ecological research. Ecological Modelling, 63: 103-112

Salski A, Kandzia P (1996) Fuzzy sets and fuzzy logic in ecological modelling. EcoSys, 4: 85-98

Salski A, Fränzle O, Kandzia P (Editors) (1996) Fuzzy Logic in Ecological Modelling. Ecological Modelling, special issue, v. 85/1

Salski A (1999) Ecological modelling and data analysis. In: H.-J. Zimmermann, Practical applications of fuzzy technologies (in: D. Dubois, H. Prade, The International Handbook of Fuzzy Sets Series, v.7), Kluwer: 247-266

Salski A, Noell Ch (2001) Fuzzy linear programming for the optimization of land use scenarios. In: N. Mastorakis et al.(eds): Advances in Scientific Computing, Computational Intelligence and Applications, WSES Press: 355-360

Shafer G, Pearl J (1990) Readings in uncertain reasoning. Morgan Kaufmann, San Mateo, CA

Silvert W (1997) Ecological impact classification with fuzzy sets. Ecological Modelling, 96:1-10

Silvert W (2000) Fuzzy indices of environmental conditions. Ecological Modelling, 130:111-119

Wieland R, Schultz A, Hoffmann J (1996) The use of neural networks and fuzzy- methods in landscape modelling. Proc. of FUZZY'96, Zittau: 370-379

Zadeh LA (1965) Fuzzy sets. Information and Control, 8: 338-353

Zadeh LA (1975) The concept of a linguistic variable and its application to approximate reasoning. Information Science, 8: 199-249

Zimmermann H-J (1993) Fuzzy Set Theory and its Applications. Kluwer Academic Publishers, Boston, Dordrecht, London, p. 399

Zheng D (2001) A Neuro-fuzzy approach to linguistic knowledge acquisition and spatial decision making. PhD-Thesis, University of Vechta, p. 156

Zhu A-X, Band LE, Dutton B, Nimlos TJ (1996) A semi-arid grazing ecosystem simulation model with probabilistic and fuzzy parameters. Ecological Modelling 90/2, 123-146

Ecological Applications of Qualitative Reasoning

B. Bredeweg · P. Salles · M. Neumann

2.1 Introduction

Most of current ecological knowledge is qualitative and fuzzy, expressed verbally and diagrammatically (Rykiel 1989). This chapter discusses an approach known as Qualitative Reasoning (QR) to formally represent and automate reasoning with that kind of knowledge. QR does not use nor require numerical data and promises to be of great importance for capturing ecological knowledge.

QR is an area of Artificial Intelligence (AI) that is concerned with the construction of *knowledge* models that capture insights domain experts have of systems' structure and their behaviour (functioning). The behavioural aspect studied most is *qualitative prediction of behaviour*, i.e. analysing how the behaviour of a system evolves as time passes. Although any system can be an object of such a reasoning process, traditionally the majority of research deals with physics and engineering (Weld and de Kleer 1990). Successful application areas include autonomous spacecraft support (Williams et al., 2003), failure analysis and on-board diagnosis of vehicle systems (Price and Struss 2003), automated generation of control software for photocopiers (Fromherz et al. 2003), and intelligent aid for learning about thermodynamic cycles (Forbus et al. 1999). Thus, QR is relevant for researchers that are interested in important AI issues as well as for managers, developers, and engineers who are looking for potential industrial benefits of AI.

As QR technology is probably new to many ecologists, this chapter discusses and explains the technology using well-understood examples from other domains in order to not confuse the characteristics of the technology with a discussion on how to represent ecological knowledge using QR. The organisation of this chapter is as follows. The section below further reviews the need for QR in ecological applications. The next section then discusses general characteristics of QR. After that, the application of QR technology for modelling population behaviour is presented, followed by a section in which current applications of QR for ecology are discussed. The chapter ends with a brief concluding section.

2.2 Why Use QR for Ecology?

The main goal of ecological research is to understand the structure and functioning of nature. The structure consists of objects, such as individuals, populations, communities, and their relations with the physical world organized in ecosystems and landscapes. Functioning is explained by imposing causal relationships on observable features of those objects in a way that it is possible to understand why things happen in ecological systems. However, ecological systems have features that put strong barriers on research and knowledge representation, including: the complexity of any ecological system, and the difficulty to obtain long-term good quality data and to run controlled experiments. Hence, ecological knowledge is heterogeneous, including both quantitative and qualitative aspects. There is need for new and efficient computer-based tools to adequately capture that knowledge. As noted by Rykiel (1989), ecologists have a considerable amount of knowledge 'in their heads' and not many ways to make this knowledge explicit, well organized, and computer-processable.

Models and simulations are important tools for ecological research. Ecologists often frame their ideas in modelling expressions and test them through simulations. There are two distinct approaches to modelling ecological systems: statistical models and structural models (Bossel 1986). Much of ecological research relies on statistical models. These models usually do not capture the available structural knowledge and the parameters of statistical models usually have no counterpart in the real system. Structural models, on the other hand, are tools for describing system structure and system elements as close as possible to real systems. This way, such models mimic real objects, structural connections, parameter values and may provide behavioural predictions grounded in the system structure.

QR models are structural models that are particularly adequate to support the understanding of behaviour of systems. The following QR features are especially attractive for modelling ecological knowledge:

- Approaches to QR provide a rich vocabulary for describing objects, situations, relations, causality, assumptions, and mechanisms of change. Using this vocabulary it is possible to capture *conceptual* knowledge about systems and their behaviour and use that knowledge to automatically derive relevant conclusions without requiring numerical data.
- QR modelling uses a compositional approach to enable reusability. This is achieved by constructing libraries of partial behaviour descriptions (model-fragments) that apply to the smallest entities relevant within a domain. As larger systems are built from these basic elements, reasoning about the behaviour of larger systems means combining the behaviour of these elements. This prevents having to develop dedicated models for each system encountered.

- QR models provide causal explanations of system behaviour. As causal relations are explicitly represented in model-fragments, it is possible to derive the behaviour of a complete system from the behaviour of its constituents and to automatically generate insightful explanations that *causally* explain the functioning of the overall systems in terms of its constituents.
- QR creates representations for continuous aspects of the world to support reasoning with little information, including incomplete knowledge or knowledge expressed just in qualitative (linguistic) terms (without using any numerical information).

Qualitative models automate *conceptual* knowledge. Being explicitly represented this knowledge can be inspected, possibly modified, by users and by other modellers. The construction of such *qualitative* models is of particular interest for education, training, management, and decision-making, because they facilitate structured expression and communication of insights among participants. After all, many questions of interest in ecology can be answered in terms of 'better or worse', 'more or less', 'sooner or later', etc. (Rykiel 1989).

2.3 What is Qualitative Reasoning?

Early work on QR focuses on automatic generation of explanations (Brown et al. 1982; Hollan et al. 1984) in the context of interactive learning environments, that is, educational software that fosters learning by having learners interact with a simulation of the subject matter. Key QR publications present approaches to having *computers* perform conceptual analysis of system behaviour (Bobrow 1984). From this work originates the idea of using *qualitative* models and simulations, also referred to as *articulate* simulations (Forbus 1988; Bredeweg and Winkels 1998). A typical QR model captures a representation of both the structural and the behavioural aspects of a system. A qualitative model abstracts from quantitative information by using an ordered set of qualitative values, usually a set of alternating points and intervals referred to as a quantity space. Quantities are assigned values from such quantity spaces, allowing quantities to capture qualitative distinct behavioural features of a system. Changing behaviour is represented using a qualitative derivative for each quantity, representing: decreasing ∂=[–], steady ∂=[0], and increasing ∂=[+]. Another typical aspect of a QR model is the explicit representation of causality. Different types of modelling primitives have been introduced in this respect, each type having a specific conceptual meaning and a formal defined calculus allowing implementation in computer programs. Following these basic ideas a wide range of topics have been tackled. To name a few: order of magnitude reasoning (Raiman 1986), alternative approaches to inferring causality (Iwasaki and Simon 1986; de Kleer and Brown 1986), reasoning with multiple models (Addanki et al. 1991; Weld 1988), compositional modelling using assumptions (Falkenhainer and Forbus 1991),

integration with numerical simulation (Amador et al. 1993), and varying granularity in the representation of time (Rickel and Porter 1997).

Using the QR representational primitives, libraries of model-fragments can be constructed that capture knowledge from a certain domain. QR engines use these libraries to automatically generate qualitative models of systems belonging to such a domain. Building a library is thus a fundamental aspect of using QR technology. In the past, considerable effort has been put in building qualitative models for the domain of physics (e.g. Collins and Forbus 1989; Kim 1993). Libraries for other domains still need to be developed and made to use. Lately, libraries capturing ecological knowledge are being created (Salles and Bredeweg 2003).

2.3.1 A Working Example

Let us consider a simple two-tank system, with tanks of equal width, for which it is known that both tanks contain a certain amount of oil and that the oil-column is higher on the left-hand side (LHS). Let us assume that the relative heights of the two tanks are unknown. Now suppose that the two tanks are connected via a pipe with a valve, placed at the bottom of the containers. When the valve closing this pipe is opened, what behaviours may happen? Figure 2.1 illustrates the answer to this question.

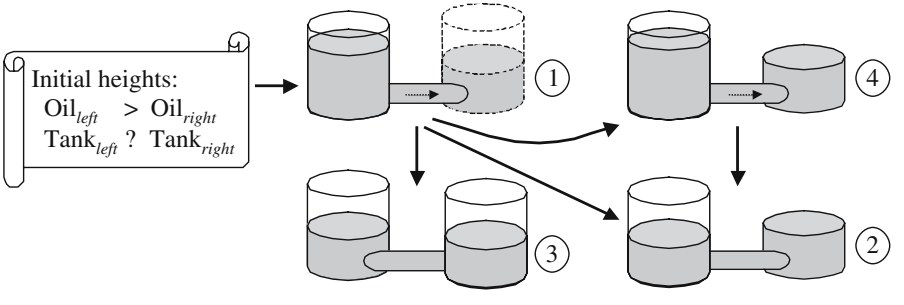


Figure 2.1. Possible behaviours of a two-tank system

The oil-column on the LHS is higher than on the right-hand side (RHS). Hence, oil will flow from the LHS into the RHS tank until the pressure-difference becomes zero and the system reaches an equilibrium. Since the initial description does not specify the relative heights of the tanks (as visualised by the dashed line in situation 1) multiple behaviours are possible. There are three qualitatively distinct possibilities. If the tank on the RHS is high enough it will be able to contain all the inflowing oil (situation 3). Alternatively, the RHS tank may at the start already be lower than the LHS oil-column. In this case, oil will be spilled (situation 4) until the height of the decreasing LHS oil-column becomes equal to

the height of the smaller RHS tank (situation 2). Finally, it may be the case that the RHS tank is smaller but still high enough to contain all the inflowing oil. The system stabilises at the moment that the RHS tank becomes fully filled (situation 2).

Notice that different behaviours would be predicted when more (or less) information is initially known. For instance, knowing the relative tank heights would result in predicting less possible behaviours. Humans are flexible in this respect. They apply the same basic knowledge to different situations producing appropriate conceptual analysis. This is also one of the features of QR and rather different from traditional approaches using numerical simulations. Instead of having a single fixed model, a QR engine automatically assembles a unique model to fit a particular situation. The sections below discuss this idea in more detail as well as other prominent features of QR. Together they show how conceptual behaviour analysis can be formalised and reasoned with automatically using QR technology.

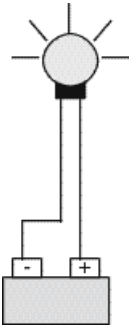
2.3.2
World-view: Ontological Distinctions

QR provides explicit representations of the conceptual modelling layer, rather than only an executable mathematical expression. This is crucial to any attempt to support and automate model building and is one of the major issues of QR. This section discusses the two main ideas that have been developed in this respect, as well as an alternative approach.

2.3.2.1
Component-based Approach

De Kleer and Brown (1984) describe a component-based approach to qualitative reasoning. In their approach the world is modelled as *components* that manipulate *materials* and *conduits* that transport materials. Physical behaviour is realised by *how* materials such as water, air and electrons, are manipulated by, and transported between, components. How components manipulate materials is described in a library of component models. In these descriptions a component is associated with qualitative equations known as confluences: relations between variables that describe the characteristics of the materials. The model of a certain component may consist of a number of qualitative-states, each specifying a particular state of behaviour. More specific, a qualitative state consists of a name, one or more specifications and a set of confluences. The specifications define the conditions that must be true for the qualitative state to be applicable. The confluences describe the specific behaviour of the materials in this state of behaviour. Figure 2.2 illustrates the basic idea. It shows a device consisting of two components, a battery and a lamp. The battery has three qualitative states of

behaviour: fully charged, partially charged, and empty. The lamp has two qualitative states: it can function normally (OK) or it is broken. The behaviour of the device as a whole is generated using the cross-product. That is, all the possible behaviours (qualitative states) of each component are combined with all the possible behaviours of all other components (see Table in Figure 2.2).



		Battery		
		Fully charged	*Partially charged*	*Empty*
Lamp	*OK*	Bright light	Soft light	No light
	Broken	No light	No light	No light

Figure 2.2. Possible behaviours of a lamp connected to a battery

Generating the cross-product and determining the consistency of each potential state of behaviour is referred to by de Kleer and Brown as the *intrastate* analysis. After this analysis, the problem is to find out which states of behaviour will be successors as time passes by. This is referred to as the *interstate* analysis, which tries to determine whether the behaviour within a certain state may lead to the termination of that state. In other words, to find out if the values of variables are changing such that they, when time passes by, no longer fall within the specifications of the overall state of behaviour. In the component-based approach this is realised by applying rules that must hold between states. An example of such a rule is the *limit rule*: if in the current state a variable has a value and increases or decreases, then it will respectively have the adjacent higher value, or the adjacent lower value, in the next state. For instance, the overall system behaviour 'soft light' may move into the behaviour of 'no light' when the battery power decreases to zero (and thus moves from qualitative state 'partially charged' to 'empty'). Another important rule is the *continuity rule*: each variable value must change continuously over states. For instance, the battery cannot immediately change from 'fully charged' to 'empty'.

2.3.2.2 Process-based Approach

Forbus (1984) describes a process-based approach to qualitative reasoning. In this approach the world is modelled as consisting of physical *objects* whose properties are described by *quantities*. Physical behaviour refers to these objects being created, destroyed, and changed. Although in principle anything can be represented as an object, there is a commitment in qualitative process theory (QPT) to represent physical objects as closely as possible to how humans perceive the physical world. Two important primitives in the process-based approach are *individual views* and *processes*. Views describe the characteristics of an object or a group of objects, e.g. of a container containing a liquid. Processes describe mechanisms of change. These changes are represented by *influences*. They describe the changes that occur when a process is active. Typical examples of processes are heat-flow and liquid-flow. The former describes energy exchange between objects with unequal temperatures. The latter describes how liquid flows between connected containers with unequal pressures. Figure 2.3 depicts the behaviour of a boiler system and illustrates the basic idea of how QPT uses *processes* and the notion of *limit-analysis* as the basis for behaviour prediction.

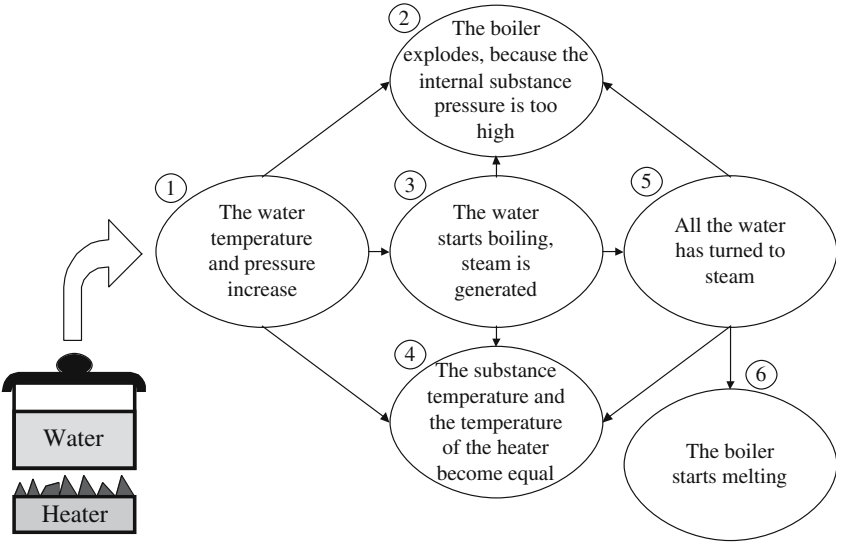


Fig. 2.3. Possible behaviours of a boiler system

The boiler system consists of a heater and a container. The container contains water and is being heated. What behaviours may occur and which processes cause them?

1. After the heater is turned on, a heat-flow process causes energy to flow from the heater to the container and the water. This causes the water

temperature, the container temperature, and the internal container pressure to increase. This behaviour may lead to three other behaviours (2, 3 or 4), due to limits being reached.

2. The boiler explodes because the internal pressure becomes too high. The reaction force generated by the container is lower than the pressure exerted by the substance it contains. The boiler system is broken after this behaviour. Hence the simulation stops here.
3. The temperature of the water reaches its boiling point. A new process 'boiling' becomes active which causes the generation of steam. This behaviour may lead to three other behaviours (2, 4 or 5), due to limits being reached.
4. The temperature of the substance in the container (be it water or steam) is now equal to the temperature of the heat source. From here on, no further changes take place.
5. All the water has now turned into steam. The boiling process has stopped, but the heat-flow continues. This behaviour may lead to three other behaviours (2, 4 or 6), due to limits being reached.
6. If the heater is warm enough it may ultimately cause the container to melt, because the container temperature will reach its melting point. The boiler system is broken after this behaviour. Hence the simulation stops here.

Notice that, despite many differences (see Bredeweg 1992) the global idea of using a library of model-fragments (albeit consisting of views and processes) is similar to the idea of using a library of component models in the component-based approach.

2.3.2.3
Constraint-based Approach

Kuipers (1986, 1994) describes the constraint-based approach. This approach takes a qualitative version of a differential equation as starting-point. The basic assumption is that Ordinary Differential Equations (ODE's) can be rewritten into Qualitative Differential Equations (QDE's). The qualitative differential equations can be used for qualitative simulation. In the constraint-based approach there is no explicit representation of entities from the (physical) world. This approach also does not use a library of any kind from which models can be assembled during simulation. Instead, the qualitative reasoning engine is provided with a description of some aspect of the (physical) world in terms of the qualitative constraints between variables as shown in Table 2.1. Notice that each qualitative constraint maps onto a specific aspect of the ordinary differential equations.

Behaviour prediction with constraint models is done by applying a kind of generate and test cycle that produces the possible behaviours of a system. The generation part determines how a state of behaviour may change into a new state of behaviour, by applying *transition* rules to each function in the current state of behaviour. Testing is concerned with determining the consistency of a certain state, by applying constraint satisfaction to the constraint model that represents the behaviour in that state.

Table 2.1. Qualitative constraints and mathematical functions

Qualitative constraints (QDE's)	*Mathematical functions (ODE's)*
ADD(*f*, *g* ,*h*)	$f(t) + g(t) = h(t)$
MULT(*f*, *g* ,*h*)	$f(t) \cdot g(t) = h(t)$
MINUS(*f*, *g*)	$f(t) = -g(t)$
DERIV(*f*, *g*)	$f'(t) = g(t)$
M+(*f*, *g*)	$f(t) = H(g(t)) \wedge H'(x) > 0$
M–(*f*, *g*)	$f(t) = H(g(t)) \wedge H'(x) < 0$

2.3.2.4 Suitability of Approaches

Although the constraint-based approach is probably one the most used approach it has some drawbacks. The main issue is that it does not support deriving behaviour from the physical structure (see also next section). Instead, it takes differential equations as a starting point. In a way, the constraint-based approach has the same drawbacks as traditional numerical approaches. The modelling primitives provided by this approach do not allow *symbolic* modelling of the conceptual knowledge that domain experts have. Notions such as processes, static properties, causality, and physical structure cannot be represented by this approach explicitly. The component-based approach does facilitate the representation of much of this kind of knowledge. However, the ontology of interconnecting components seems more suitable for human made artefacts than for natural systems. From an ontological perspective, QPT is probably most suitable for building models about ecological systems (Salles 1997).

2.3.3 Inferring Behaviour from Structure

In general, a qualitative reasoning engine takes a *scenario* as input and produces a *state-graph* capturing the qualitatively distinct states a system may manifest

(Figure 2.4). A scenario usually includes a structural description of the physical appearance of the system. Such a description models the entities (e.g. physical objects and components) that the system consists of, together with statements concerning the structural organisation of these objects (e.g. a container *containing* liquid). Often a scenario also includes statements about behavioural aspects such as relevant quantities and (in-)equality statements between some of those quantities.

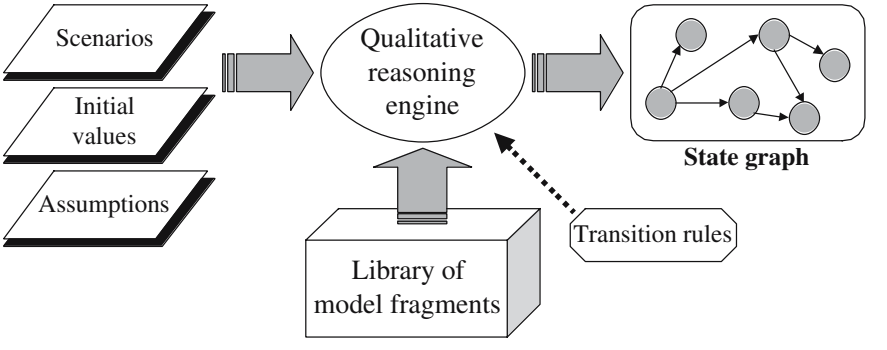


Figure 2.4. Basic architecture of a qualitative reasoning engine

A state-graph consists of a set of states and state-transitions. A state refers to a qualitatively unique behaviour that the system may display (a possible state of behaviour). Similar to a scenario, a state consists of a set of declarative statements that describe the physical structure of the system and the behaviour it manifests at that moment. A state is typically characterised by a set of qualitative values of relevant quantities representing their magnitude and direction of change. A state-transition specifies how one state may change into another state. A sequence of states, connected by state-transitions, is called a behaviour-path, but is also referred to as a behaviour trajectory of the system. A state-graph usually captures a *set* of possible behaviours-paths, because multiple state-transitions are possible from certain states. To further detail these notions, consider again the two-tank system from Figure 2.1. The simulation results obtained from a qualitative model of this system are shown in Figure 2.5[1]. The state-graph (LHS) shows the four possible states that the two-tank system may manifest. Each black circle refers to a possible behavioural state, the state numbers refer to identifiers created by the reasoning engine[2], and the arrows indicate which states may succeed each other.

[1] The diagrams are generated by VISOGARP (Bouwer and Bredeweg 2001)

[2] Notice that the numbers are identifiers created by the reasoning engine and that they do not necessary reflect the order in which states of behaviour occur.

Thus, the conditions set in the scenario (referred to as 'input' in Figure 2.5) lead to state 1. This means that (with the knowledge the engine has) there is a unique interpretation of the scenario. From state 1 three successors are possible: 4, 2, and 3. Apparently, there is ambiguity concerning the possible transitions. If state 4 occurs it is always followed by the behaviour represented by state 2. States 2 and 3 have no successors, they are end-states, and represent an equilibrium of some kind.

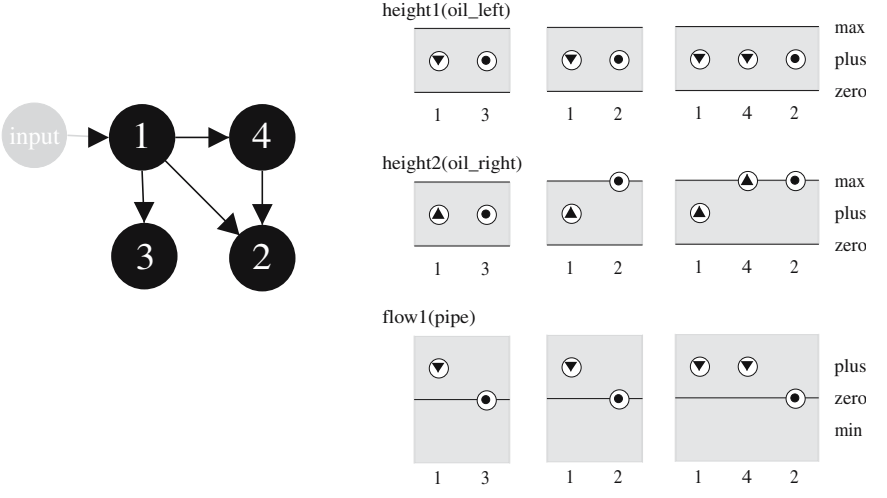


Figure 2.5. Simulation results of a model for the two-tank system

Notice that a QR engine generates all possible solutions. That is, given a scenario (a structural description) it will generate *all* behaviours that are consistent, and thus possible to infer, with the details defined in that scenario. This is rather different from a numerical simulation that usually produces one specific answer. One of the interesting features of QR is the awareness it creates for all possible interpretations of a certain situation. This can for instance be useful to support management tasks. The results obtained by the simulation show all that may happen. If certain behaviours are not acceptable, preventive actions can be taken.

2.3.4 Qualitativeness and Representing Time

Qualitative prediction of behaviour is concerned with reasoning about the properties of the physical world that change over time. Particularly, to include only those qualitative distinctions in a behaviour model that are essential for solving a particular task for a certain system. The goal is to obtain a finite representation that leads to coarse, intuitive representations of systems and their

behaviour. Central to qualitative reasoning is thus the way in which a system is described during *a period of time in which the qualitative behaviour of the system does not change*. The notion of change is subtle, because numerical values of variables may change whereas from a qualitative point of view the behaviour of the system remains constant. During a heat-flow process, for example, the temperature of a liquid may increase, but from a qualitative point of view it is still a liquid, until another process (boiling) becomes active and the liquid becomes a gas.

In QR the representation of time is closely intertwined with the representation of quantity values. Changes in the values of quantities represent time passing. The possible qualitative values of a quantity may be divided into points and intervals. A quantity can therefore, during a certain period of time (of constant qualitative behaviour), have its value either at a point or at an interval. The intuitive understanding behind this approach is illustrated in Figure 2. 6 for the quantity temperature as it is used to describe the physical state of a substance.

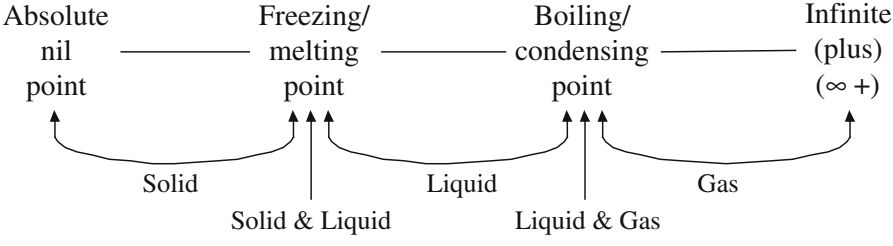


Figure 2.6. The quantity space for the temperature of a substance

All the quantitative values a substance temperature can have are divided into six qualitative values, consisting of three intervals and three points. Each value resembles a characteristic period of constant qualitative behaviour for the substance. If, for example, the temperature has a quantitative value somewhere between freezing point and boiling point and this value increases, then the substance shows constant qualitative behaviour, namely 'being a liquid', until it reaches its boiling point. As soon as it reaches this boiling point, the substance arrives at a new time interval in which it again shows constant qualitative behaviour, namely boiling.

In qualitative models this knowledge is formalized as follows. A quantity value is represented as the pair <Magnitude,Derivative>. Magnitude represents the amount of a quantity and the Derivative represents the direction of change over time. The values a Magnitude can take on are represented in a Quantity Space (QS). Consider again the two-tank system (Figure 2.1). The amount of substance in a tank can be represented as having three possible magnitudes: QS={zero,plus,max}, respectively meaning there is no substance, there is some substance, and the amount of substance in the container has its highest possible value: maximum. Values for the Derivative are also represented by a quantity space, namely QS={min,zero,plus}, meaning the Magnitude is decreasing, steady,

and increasing. Thus, if amount has the value *amount*=<*plus,plus*> this can be read as: there is an amount and in the current state it is increasing.

A *value-history* diagram shows the qualitative values generated by a QR engine. Figure 2.5 (RHS) shows the value-history for the quantities[3] involved in the model of the two-tank system for all the behavioural states. For instance, in state 1 the *height* of the *oil_right* has magnitude *plus* and is *increasing*, hence: <*plus, plus*>. Next, in state 4 this quantity has the value *max* and is still *increasing:* <*max, plus*> (representing overflow). Finally, in state 2, it has again value *max*, but now it is steady: <*max, zero*>. This can be inferred from the diagram (Figure 2.5) as follows. The possible values that *height* can take on are shown on the RHS: QS={zero, plus, max}. The circles above the state numbers designate the specific value the quantity has in that state. In addition, the circles contain a small arrow pointing up, or down, or a small black circle. These indicate that the quantity is increasing, decreasing, or steady, respectively. A sequence of quantity values is referred to as a value-history and it follows a behaviour-path. In the case of the two-tank system there are 3 behaviour paths: [1 → 3], [1 → 2], and [1 → 4 → 2].

Determining the relevant quantity space for each quantity is an important aspect of constructing a qualitative model because it is one of the features that determines the variety of possible behaviours that will be found by the engine when the model is simulated. Inequality statements (e.g. *height oil_left > height oil_right*) are also important in this respect. In fact, each qualitative distinct state of behaviour is defined by a unique set of values and inequality statements. Transitions between behavioural states are the result of changes in these values and inequality statements. State transitions are shown in a state-graph as arrows connecting the circles (Figure 2.5). For example, while going from state 1 to state 4, the magnitude of *height* (for *oil_right*) changes from *plus* to *max*. Going from state 4 to state 2 the oil heights in the two tanks become equal (not shown in Figure 2.5) and the *flow* becomes *zero*. In addition, the heights for both columns stop changing (∂=0).

2.3.5 Causality

Analyzing and explaining the behaviour of a system in terms of cause-effect relations is central to human reasoning and communication. When we think that 'A causes B', we believe that if we want B to happen we should bring about A, and if B happens, then A might be the reason for it. Causality can also be perceived as being indirect: 'A causes C indirectly' if 'A causes B' and 'B causes C'. Formalizing the notion of causality and exploiting it in automated reasoning is the basis for explanation facilities in QR systems. QPT explicitly distinguishes

[3] The model also includes the quantities amount and bottom-pressure for each tank, but these are not shown in the figure.

between changes that are caused *directly* or *indirectly* (Forbus 1984). Forbus refers to this as the *causal directness hypothesis*: changes in physical situations are caused by processes (*influences*, represented as {I+, I-}), or by propagation of those direct effects through functional dependencies (*proportionalities*, represented as {P+, P-}). This hypothesis puts three further constraints on how influences and proportionalities should be applied. Firstly, all changes are initialised by influences. Without an influence, or for that matter a process, there is no change and therefore no behaviour in the physical world. Proportionalities are used to propagate changes, introduced by influences, throughout the whole system. Secondly, both influences and proportionalities are *directed*, i.e. their effect propagates in one direction only. The influencing quantity has to be known before the dependent quantity can be determined. The relations may not be used the other way around, because this would violate the causal chain of changes, which is one of the essential features of QPT. Thirdly, no quantity may be influenced directly and indirectly simultaneously. According to Forbus, a physics that allows a quantity to be influenced both directly and indirectly at the same time must be considered inconsistent, because it also violates the essential, non-recursive, chain of causality.

Both direct influences and qualitative proportionalities are modelling primitives that express causal relationships between quantities, and have *mathematical* meaning. Direct influences determine the value of the derivative of the influenced quantity. For example, the relation I+(Y,X) means that dY/dt = (... + X ...). By definition, the quantity X is a *rate* and its value should be added to Y. Qualitative proportionalities carry much less information than direct influences. For example, the relation P+(Y,X) means that there is some monotonic function (f) that determines Y, and Y is increasing in its dependence on X, such that Y = f(... X ...) and dY/dX >0. A quantity that is not influenced by any process is considered to be constant. Notice that a single direct or indirect influence statement does not determine, by itself, how the quantity it constrains will change. Its effect must be combined with all the active influences on that quantity. Ambiguities may arise when positive and negative influences are combined and their relative magnitudes are not fully known. In these cases, the reasoning engine either considers all the possible combinations or any explicitly represented assumption that may constrain the system's behaviour.

Figure 2.7 shows a subset of the dependencies that hold in state 1 of the simulation of the two-tank system model (Figure 2.1). Such a set of dependencies is often referred to as the causal model. The diagram shows that the two oil-columns have unequal *heights* and (bottom) *pressures*. The *flow* (rate) between the two tanks depends on the difference between those pressures (and is qualitatively proportional to it). The flow has a negative influence on the oil-column with the higher pressure and a positive influence on the other, decreasing and increasing the two amounts of oil respectively. Changes in the amounts propagate to changes in heights, which in turn change the pressures. Notice that this diagram also shows the quantity space for each quantity, the current value, and the direction of change. The latter is visualised by triangles pointing up (increasing), or down (decreasing), and by small black circles (steady) (as no quantity is steady in state 1, circles are

not shown in Figure 2.7). The direction of change icon is placed adjacent to the current value of the quantity, highlighting the latter in the context of its quantity space. For instance, for the *oil_left* holds: *height*=<*plus,min*>.

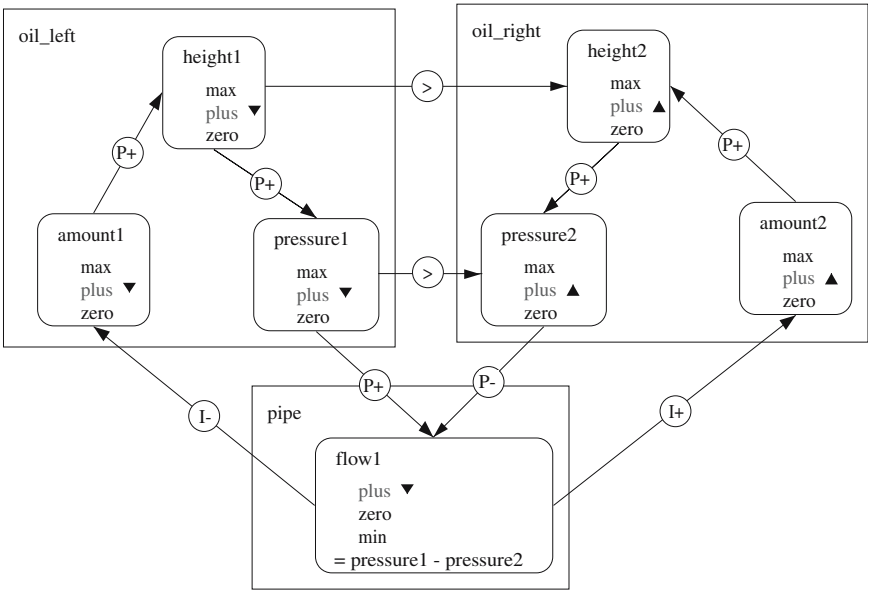


Figure 2.7. Causal dependencies for the two-tank system

It is relevant to mention that QR models represent changes in the causal structure that may happen during the simulation. For example, when the heights in the two tanks become equal (in Figure 2.5, states 2 and 3) the flow stops and the direct influences resulting from the process no longer exist. Therefore, the causal model of the two-tank system in states 2 and 3 is different from the one shown in Figure 2.7 (because the inequality sign will be replaced by an equality sign, and the arrows representing I+ and I- will be deleted). Being able to change the causal model is an important feature of QR.

The notion of causality is complex and competing ideas exist about how to capture it in models, such as the notion of causal ordering (Iwasaki and Simon 1986). However, the approach taken by QPT is generally seen as a principled one. It also seems most promising to as a means to capture causal reasoning in the domain of ecology (Salles 1997).

2.3.6
Model-fragments and Compositional Modelling

Most QR systems aim at building libraries of elementary, context independent model-fragments (component behaviour, processes, etc.). This provides the basis for automating the model composition and for the re-use of models, a highly desirable feature both for theoretical development and for applications. Model-fragments can thus be seen as re-usable conditional statements that capture knowledge about the phenomena existing in a certain domain. Model-fragments applicable to a scenario are assembled by the engine and used to infer the behaviour of the system specified in that scenario (Figure 2.4). They are also used to infer the facts true in each of the successor states. This implies, among other things, that the set of facts may change and can be different for alternative states. In general, a model-fragment requires certain structural details to be true (e.g., a tank, a liquid and a contain relation between these two entities). If the required structure exists the model-fragment is instantiated for that structure and introduces the behaviour details that apply to it (e.g., the quantities amount, height, pressure and the dependencies that hold between them). A specific model-fragment can be instantiated multiple times, namely for each occurrence of the structure to which it applies.

Preferably, model-fragments implement the 'first principles' (the fundamental laws) relevant to a domain, enhancing their usability across different systems. Reusability requires that model-fragments represent behavioural features independent from the specific environment in which they operate. De Kleer and Brown (1984) discuss a set of modelling principles for realising this objective. One principle is the 'no-function-in-structure', which states that the model of a specific component may not presume the functioning of the device as a whole. For instance, the qualitative states of a lamp (Figure 2.2) may not specify 'lit' or 'not lit', because 'being lit' depends also on the battery (and not just on the lamp). A properly functioning lamp will still not produce any light when the battery is empty or not connected to the circuit. The no-function-in-structure principle is general and applies to any approach to QR that uses a library of model-fragments. Given a sufficiently well developed library for a certain domain the qualitative reasoning engine can predict the behaviour of all kinds of systems belonging to that domain.

2.4
Tools and Software

The availability of software and tools to construct and simulate QR models is limited. QPE (Forbus 1986) is a reasoning engine that implements QPT, but using this package requires programming skills in LISP. QSIM (Kuipers 1986) is the implementation of the constraint-based approach and can be downloaded from the

WWW[4]. Recently easy to use QR model-building learning environments have been developed, notably Betty's Brain (Biswas et al. 2001) and Vmodel (Forbus et al. 2001). These packages are used for teaching in middle schools and are optimized for that purpose. Although useful in classroom situations, essential features of QR are missing and hence these tools are limited in their potential to capture expert knowledge. To further discuss the use of QR for capturing ecological knowledge, this section focuses on the toolset Homer (Bessa Machado and Bredeweg 2003), Garp (Bredeweg 1992), and VisiGarp (Bouwer and Bredeweg 2001). These tools are implemented in SWI-Prolog[5] and can be downloaded from the WWW[6]. Homer provides a graphical approach to modelling. Models created with Homer can be simulated using VisiGarp, which provides a graphical environment on top of Garp to run and inspect QR models.

2.4.1 Workspaces in Homer

Homer provides nine workspaces for creating model ingredients, divided into two categories. Building blocks are used to define ingredients (types) that can be re-used and assembled into constructs.

- Building blocks
    - *Entities*: represent physical objects or conceptualizations that are part of the system to be modelled. They form an important backbone to any model that is created. Entities are organized in a *subtype hierarchy*.
    - *Agents*: represent external influences enforced upon a system. They are thus exogenous to the system. For instance, the sun providing energy to ecological systems.
    - *Assumptions*: are labels that can be used to hide or show certain detail in a model. Typical examples are *operating* and *simplifying* assumptions. The notion of an open versus a closed population (migration or no migration) is an example of an operating assumption for models in ecology. It provides a certain perspective on the simulation. A simplifying assumption typically reduces the simulation complexity. For example, to consider a particular quantity value constant.
    - *Attributes*: define properties of entities that do not change (static). An example could be to represent the *colour* of an animal's fur as having value *brown*.

[4] http://www.cs.utexas.edu/users/qr/QR-software.html
[5] http://www.swi-prolog.org/
[6] http://hcs.science.uva.nl/projects/GARP/

  - *Configurations*: are commonly called structural relations. Structural relations model how entities are physically (or structurally) related to each other. For example: representing that a certain species is *part of* an ecological niche.
  - *Quantities*: represent changeable properties of entities and are typically seen as implementing the behavioural characteristics of a system.
  - *Quantity Spaces*: represent the values that quantities can take on.
- Constructs
  - *Scenarios*: describes the initial situation of the system whose behaviour is to be captured by the qualitative model. A scenario is the starting point for running a simulation and is created by defining (and relating) instances of building blocks. (In)-equality statements can also be added.

Model-fragments: define behavioural features for one or more entities. Model-fragments are assembled form building blocks, have conditions (specifying their applicability) and consequences (new knowledge that is true when the fragment applies), and are organized in a subtype hierarchy. Different types of model-fragments exist, notably *static*, *agent*, and *process*. An important aspect of a model-fragment is the specification of causal knowledge in terms of influences (only in processes and agents), proportionalities, and correspondences. (In)-equality statements are also defined in fragments.

2.4.2 Building a Population Model

As example consider the behaviour of a population consisting of frogs, that is only determined by 'natural' mortality and natality. Figure 2.8 shows some of the workspaces and model ingredients that may be defined for such a model.

The assumption hierarchy is shown on the LHS. Among others it defines an operating assumption for *open* and *closed* populations. The entity hierarchy is shown on the RHS. It is kept simple for reasons of clarity. A distinction is made between *biological entity* and *set of entities*. According to the model there are two kinds of biological entities (*animal* and *plant*), and there are three kinds of animals. Notice that the representation of entities follows an inheritance structure. For example, all facts that can be inferred for *animal* also apply to *frog*. A scenario is shown in the middle of Figure 2.8. It specifies the existence of a *population* named *population*[7]. This population consists of *frog* named *frog* and

[7] This is not the place to explain all the visual details in Homer. However, the fact that labels such as 'Population' appear twice has a meaning. The italic version refers to the *type* as specified in the entity hierarchy, where as the bold version refers to the *instance* name given in a specific situation. The user (the creator of the model) must provide the instance name. When omitted, Homer inserts the default name similar to the type.

has the quantity *size*. *Size* can take on five values QS={zero,low,normal,high,max}, and currently has the value *normal*. The derivative of *size* (shown by two arrows and zero) is unknown (no value is pointed out) and there is an assumption named *closed population* (identified by a question mark).

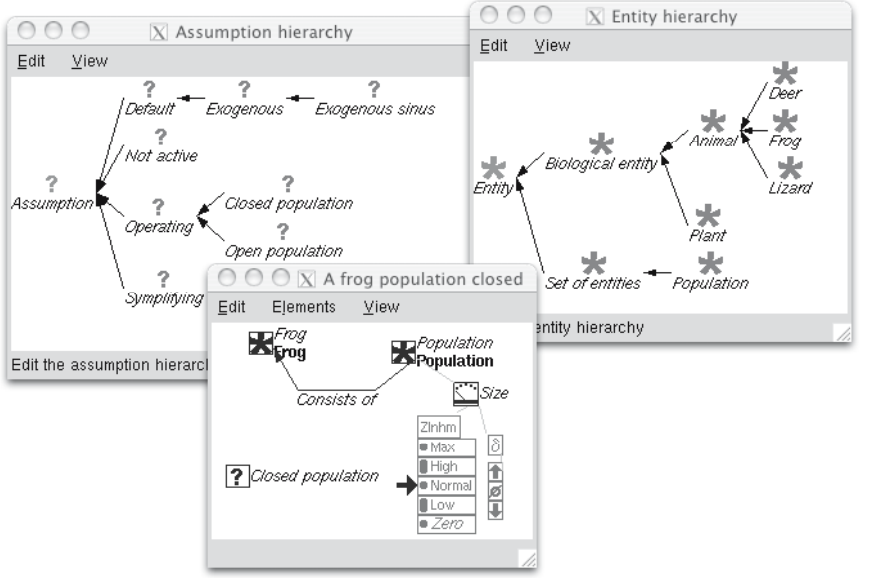


Figure 2.8. Assumption and entity hierarchy, and a scenario of a frog population model

In order for an engine to infer behaviour it needs a library of model-fragments capturing general knowledge about population dynamics. A part of this is shown in Figure 2.9. The domain theory implemented here is a qualitative reading of the equation:

Size(t+1) = Size(t) + (Born + Immigrated) – (Dead + Emigrated)

Size stands for the number of individuals of a population. *Born*, *dead*, *immigrated*, and *emigrated* refer to the amount of individual being added or removed due to natality, mortality, immigration and emigration processes. Following the QPT ontology, the representation for the four basic population processes and their effects on *Size* becomes:

I+(Size, Born); I–(Size, Dead); I+(Size, Immigrated); I–(Size, Emigrated)

The domain theory should also include feedback loops that represent the effect that *size* has on *born*, *dead*, and *emigrated*. This is obtained by means of qualitative proportionalities:

P+(Born, Size); P+(Dead, Size); P+(Emigrated, Size)

This way, the combination of *I+(Size, Born)* and *P+(Born, Size)* reads as 'the amount of individuals being born should be added to the size' and 'when the

population size changes (increases or decreases) the amount of individuals being born also changes in the same direction'. *Immigration* is not included in this feedback loop, because it is considered exogenous to the system. That is, the amount of inflow resulting from immigration does not depend on the population size. Instead, it is seen as an external factor that is determined outside the scope of the system.

These ideas are diagrammatically represented in Figure 2.9, using the workspace for defining model-fragments in Homer. The model-fragment *closed population* (LHS) is a subtype of *population* (the latter has to be included in the model before the former can become active). This previously defined fragment introduces an instance of the entity *population* (named *population*) and the quantity *size* with a five-valued quantity space (all coloured green). The assumption *closed population* (no migration) is an additional condition (coloured red). The new knowledge added by the model-fragment *closed population* includes (coloured blue): quantities *born* and *dead* (both with the two-valued quantity space QS={zero,plus}), positive proportionalities ($\propto+$) from *size* to *born* and *dead*, and bi-directional correspondences between the values *zero* for *size*, *born* and *dead* (to express the idea that when size is zero that the other quantities must also be zero).

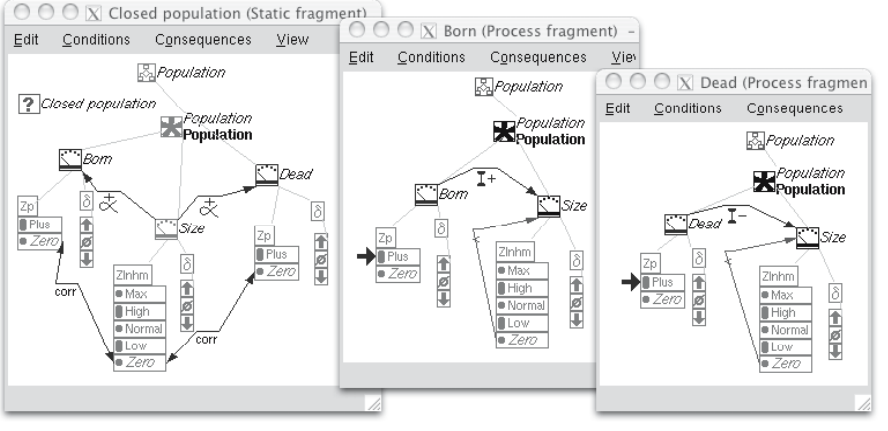


Figure 2.9. Model-fragments for a simple population model

The model-fragments *born* (Figure 2.9, middle) and *dead* (Figure 2.9, RHS) are both of type process. They require the model-fragment *population* (coloured red) to exist, before they may become true. In addition, there is an inequality statement specifying that the *size* of the population is 'greater than zero'[8] (coloured red). In other words, only if a population has some individuals these processes become active. The knowledge introduced by the model-fragment *born* is: the quantity

[8] All lines with an arrow should be read following the direction of the arrow. In this case, the inequality should be read as 'zero < (current) Quantity (value)'.

born, with value *plus* currently assigned, and a positive influence (I+) from this quantity on *size*. The value *zero* represents that there is no natality, the value *plus* means that individuals are being born. In summary, this model-fragment captures the idea that natality becomes active as soon as there is a population with some individuals, and that the number of individuals being born increases with the size of the population. The model-fragment *dead* is essentially the same except it introduces a negative influence on the population size (I–). Thus, when a population exists there will be individuals dying, which reduces the population size.

2.4.3 Running and Inspecting Models with VisiGarp

Starting with the scenario (Figure 2.8) in which the value of *size* is *normal* and the derivative is unknown (<*normal,?*>) and the library of model-fragments (Figure 2.9), the reasoning engine builds up a complete simulation model that produces the results shown in Figure 2.10. The resulting causal model in state 1 is depicted at the LHS (top). It shows the two influences from *born* and *dead* on *size* and the feedback via the proportionalities. *Size* has value *normal* and is decreasing. *Born* and *dead* are both *plus* and decrease (because of the feedback from *size*). As mentioned above, different states may show different sets of dependencies. For instance, the processes natality and mortality are not active in state 8, because *size* has value *zero*.

The state-graph is depicted at the LHS (bottom). It shows that the following behaviours are possible [1 → 6 → 8], [1 → 7], [2], [3 → 4], and [3 → 5 → 9]. The matching quantity values are depicted in the value-history (RHS). Given that in the initial scenario *size* has magnitude *normal* and derivative unknown, the interpretation of the competing influences from *born* and *dead* leads to three states. In state 1 the population decreases (*born* < *dead*), in state 2 it remains stable (*born* = *dead*), and in state 3 it increases (*born* > *dead*). The behaviour captured in state 1 proceeds via state 6 to state 8, in which the population has become extinct. Alternatively, state 1 may proceed to state 7, in which case the population stabilises. In a similar way the increasing behaviour captured in state 3 may proceed to state 4, and stabilise, or further increase via state 5 and reach the maximum value in state 9. In summary, if both *born* and *dead* are positive, but their relative magnitudes are unknown all behaviours are possible. The population can grow to its maximum size, go extinct, or stabilise at any intermediate value.

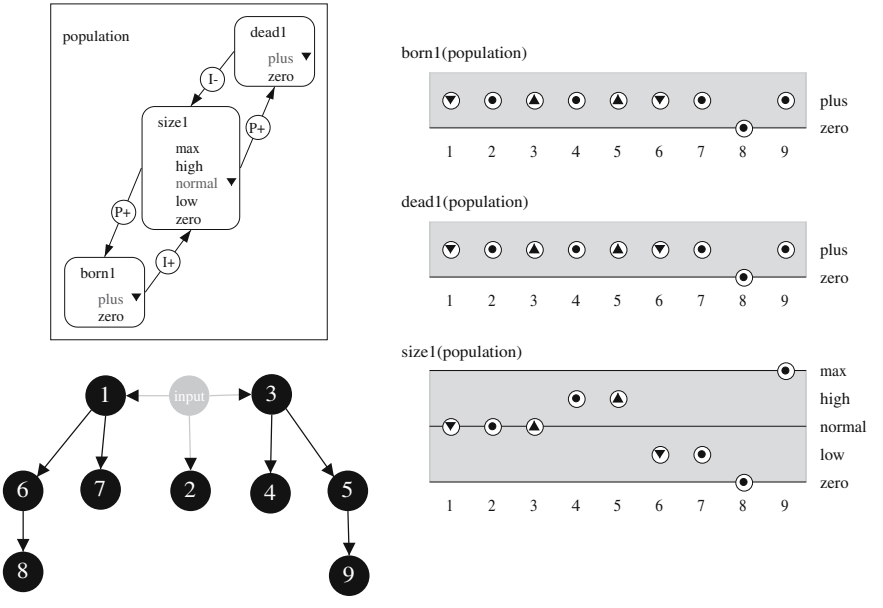


Figure 2.10. Simulation results for a closed population model

2.4.4
Adding Migration to the Population Model

The model constructed so far assumes a population without migration. Hence the population cannot recover from extinction and in the simulation there is no transition from state 8 in Figure 2.10. Figure 2.11 depicts three of the model-fragments needed to implement migration. The *open population* model-fragment (LHS) competes with the *closed population* discussed above (Figure 2.9). It applies when the assumption *open population* is true (thus, when specified in a scenario).

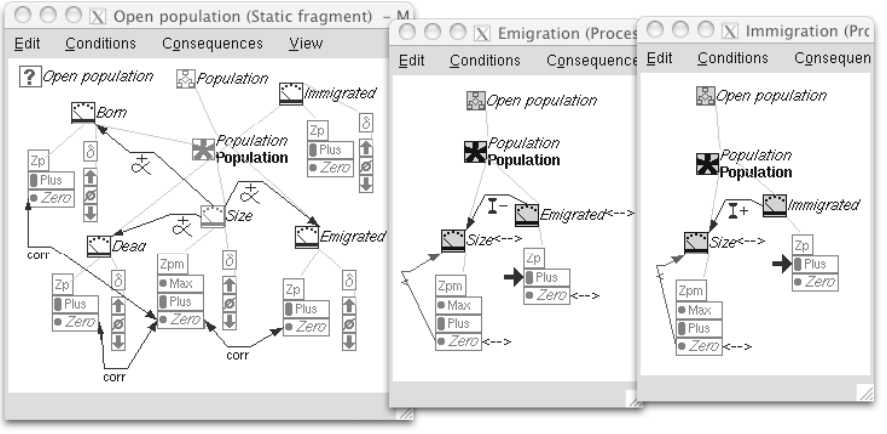


Figure 2.11. Model-fragments for a population model including migration

In addition to the quantities *born* and *dead* it introduces *emigrated* and *immigrated*. Changes in population size also affect emigration, but not immigration. Next, a set of model-fragments is required to implement the migration processes. *Emigration* and *immigration* are shown in Figure 2.11 (middle and LHS). Similar to *dead* and *born* they start when a population exists. When active, *emigration* has a negative influence on the population size and *immigration* has a positive influence. *Colonization* is modelled as a special case of *immigration* (not shown in the Figure). Descriptions of these two processes are similar, but *colonization* starts a new population where such a population does not exist. Hence, population *size* has to be *zero*.

A simulation with this extended model is shown in Figure 2.12. For the sake of clarity, *size* has been given a three-valued quantity space QS={zero,plus,max}. Also the causal model taken from state 9 (Figure 2.12, LHS, top) does not show all the available information. For instance, correspondences are not shown. Finally, the value-history shows a *particular* behaviour path, namely [13 → 14 → 18 → 9 → 11 → 15 → 1→ 13] (and not all values from all states). Notice, that this path implements a loop. In the initial scenario, the population *size* has value *plus* and an unknown derivative. The state-graph shows that nine states of behaviour match that initial description (states with numbers 1 through 9). State 9 is on the selected path and the value-history shows that the population is increasing, along with all the processes. This behaviour moves on to state 11, in which the population reaches its maximum size and stops growing. Next, the population may start to decrease (state15), reach the next lower value *plus* (state 1), and then become extinct (state 13). Colonisation may then start (state 14), and create a new population which starts gaining size in state 18, and actually 'exists' in state 9.

The transitions [13 → 14] and [11 → 15] are special from a QR point of view, as they do not reflect a value change due to increasing or decreasing. Instead, the

derivative of *immigration* changes, from ∂=[0] to ∂=[+] and from ∂=[0] to ∂=[–], respectively. This is the result of a special feature implemented into the Garp reasoning engine by means of which exogenous quantities can be assigned certain 'behaviour'. As if the external world behaves in a certain way. In this case, *immigration* has been assigned 'exogenous sinus' (Figure 2.8) by specifying this assumption in the scenario. 'Exogenous sinus' can be used to enforce a continuous change on an exogenous quantity. As a results, *immigration* changes following the pattern: ∂I=[0] → ∂I=[+] → ∂I=[0] → ∂I=[–] → ∂I=[0] → ∂I=[+] → etc. Assigning some default behaviour to exogenous quantities is required for representing population dynamics in order to enforce a disturbance of some kind through the system.

The examples discussed above present initial ideas on how to capture ecological knowledge using QR. Of course, many additional details can be represented yielding more advanced models and simulations that deliver important conclusions and explanations. The next section discusses examples of such applications.

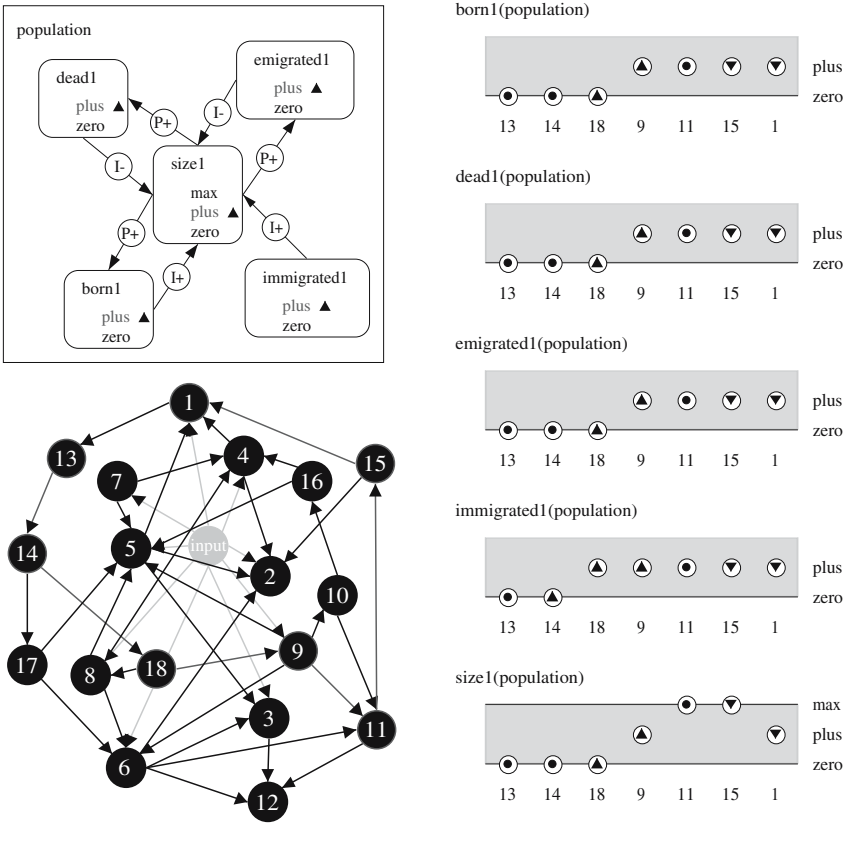


Figure 2.12. Simulation results with an open population model

2.5 Examples of QR-based Ecological Modelling

This section presents applications of QR techniques to various ecological problems. It is not the intention to be exhaustive, but to present an overview of the possibilities QR has to offer for ecological modelling. In fact, formalizing qualitative ecological knowledge in qualitative terms is a longstanding problem in ecological modelling. May (1973) performed a qualitative analysis of the results produced by differential equation models about interactions between populations to study the relationship between complexity and stability of biological communities. May used only the signs {+, 0, –} and showed that a less complex community met the conditions for stability, while the more complex was not stable. Therefore, the 'common-sense wisdom' that more complexity means increased stability may not be true.

An approach for building qualitative models about the dynamics of communities subject to recurrent disturbance (such as fire) was proposed by Noble and Slatyer (1980). This approach is based on a small number of attributes of the plant's life history (vital attributes) that can be used to characterise the potentially dominant species in a particular community. Simulations produce a replacement sequence that depicts the shifts in composition and dominance following a disturbance. Further developments describe a simulation model that is also based on the vital attributes but now combined with quantitative knowledge about the abundance of the populations and their survival according to the availability of environmental resources (Moore and Noble, 1990; 1993).

Câmara et al. (1987) describe SLIN, a program that supports qualitative simulations using values expressed in linguistic terms (such as low, medium, high) manipulated by a set of logical rules. SLIN was used in studies about the management of water resources of a hydropower plant and assessment of oil dispersion in the sea after a tanker accident (Antunes et al. 1987). Recently McIntosh (2003) describes a modelling language for dealing with partial and imprecise ecological knowledge. Borrowing some concepts from QR, such as the representation of quantities (including the distinction between amount and derivative, both having two value components, magnitude and sign, and a set of possible qualitative values), the author implements his ideas using a rule-based approach and presents an example about vegetation dynamics.

2.5.1 Population and Community Dynamics

Following a principled approach to QR Salles and Bredeweg (1997) have developed a library of model-fragments that can be used to construct models and automate reasoning about the behaviour of populations (the examples presented in section 4 are based on this work). This library was used to construct a model of

the Cerrado Succession Hypothesis (CSH) (Salles and Bredeweg 2005). The Cerrado is the second largest Brazilian biome, a kind of savannah type of vegetation with a wide range of natural physiognomies, spanning from open grasslands to closed forests. Fire is one of the most influential determinants of this physiognomy and its influence is expressed by the follow hypothesis: if the fire-frequency increases (for example, because of human actions), then the vegetation becomes less dense, with reduction of trees and shrub populations so that grass may dominate. If, on the contrary, fire-frequency decreases, the vegetation becomes denser, with more trees and shrub and less grass. A set of model-fragments defines the different physiognomies according to the proportion of three populations, tree, shrub and grass. For example, the *cerradão* is a forest defined by the maximum size of tree population and no grass. The *campo limpo* is open grassland defined by the maximum size of grass population and no trees and shrubs. Between these two extremes, other physiognomies may have more or less of the three populations. According to the literature and Brazilian researchers, it is 'commonsense' that fire destroys the litter and, under this condition, temperature and light increase and humidity decreases. These are negative influences for the germination of trees and shrub seeds, and positive influence for the germination of grass seeds. These ideas are the basis for the causal model captured in the CSH. Simulations with the CSH model produce the behaviour predicted by the hypotheses mentioned above. Notice that, the CSH is a typical situation in which a mathematical approach is not adequate, because the ecological system is complex and numerical data about the *whole* phenomena do not exist. There is 'only' a conceptual model, that is, the expert's commonsense understanding and hypothesis to explain the final result.

The work on the CSH has been the inspiration for a number of additional research efforts, among which the interactions between populations of different species. Such interactions are important for understanding the behaviour of larger communities. Salles et al. (2003a) present a set of models about interactions such as predation/parasitism, commensalism, cooperation/mutualism, amensalism, and competition. Each model produces simulation results that are characteristic for the interaction type it models. For example in the case of predation the state-graph shows four behaviours: only the prey reaching maximum size, both species stabilising at a corresponding size, both disappearing, and the predator disappearing while the prey grows to its maximum size.

The ants' garden is an interesting example of interacting species. This system, a well-known symbiosis between ants (*Formicidae*) and a fungi (*Lepiotaceae*), is more complex than initially understood. A third species, the specialized garden parasite fungi (*Escovopsis*), is often present and may destroy the system by attacking the cultivated fungi. However, it almost never happens because ants carry on their body colonies of bacteria (*Streptomyces*) that produce antibiotics specifically targeted to suppress the growth of *Escovopsis*. Traditional modelling approaches, based on differential or difference equations, are not adequate to handle this complex balance of interactions, but qualitative models can and have been made. Using the set of interacting population models Salles et al. (2003b) describe the ants' garden as follows: ants and *Lepiotaceae* fungi as mutualism;

Escovopsis and *Lepiotaceae* fungi as parasitism; ants and bacteria as commensalism; and bacteria and *Escovopsis* fungi as amensalism. One of the typical simulations with this model produces the following four behaviours: coexistence of all the involved species, complete extinction of the garden, coexistence with *Escovopsis* but the ants and *Lepiotaceae* fungi reaching their maximum size, and the elimination of the parasite, with the garden reaching its maximum size. As the authors argue, this is another example of how QR models formalises conceptual knowledge, in this case representing alternative hypotheses of systems' behaviour.

Nuttle et al. (2004) describe models to support learning and research on food chains and the trophic cascades. They present an evaluation of three alternative mechanisms for implementing the basic trophic interaction, and discuss their potential to serve as basic building blocks for building more complex representations of food chains and food webs.

2.5.2 Water Related Models

Aquatic systems offer the integration of physical, chemical and biological aspects that might be combined with social, cultural and economic aspects. Salles et al. (2003c) describe a model developed for understanding stream ecosystems, to predict values of variables and to combine such understanding with restoration and proactive actions of management. The models show the effects of good and bad management practices on the effects of pollution by organic matter and the consequences for the amount of dissolved oxygen and fish stocks. Problems found during the modelling effort and implemented solutions are discussed, including the explicit representation of assumptions and the role of ambiguities in the outcomes of the models (Salles et al. 2003c).

A model for supporting stakeholders and decision makers to address problems related to nutrient cycle in stream ecosystems is presented by Neumann and Bredeweg (2004). The model explores the concept of the spiralling of resources in segments of a river from the perspective of processes within the nutrient cycle represented by the uptake rate (from nutrients to autotrophs), retention rate (from autotrophs to detritus), and release rate (from detritus to nutrients). Each segment of the river can be characterized with the definition of attributes and the influences coming from the catchments area.

Benthic macro-invertebrate communities, which have distinct responses to physical, chemical, and biological disturbances, are particularly interesting for assessing impacts of conversion of natural landscapes to urban and agricultural uses. However, modelling is difficult in this context because information relating anthropogenic activities to benthic communities is fragmented and temporally inconclusive. Tullos et al. (2004) present models that describe the impacts of watershed development and riparian deforestation activities on benthic macro-

invertebrate communities based on a comprehensive understanding of the underlying processes that control these communities.

It is known that changes during the salmon development depend upon the moving sum of average daily water temperatures. Guerrin and Dumas (2001a,b) describe models for assessing the impact of the environment on salmon population dynamics. The models are implemented in QSIM and represent the functioning of spawning areas of salmon (salmon reeds) and the impact on mortality rates at early stages. The model consists of two sub-models that are quite complex, combining processes that occur at different time scales (fast and slow). A qualitative autonomous clock allows for the accumulation of degree-days from average water temperatures. The two sub-models are coupled via some shared variables and by means of transition states, in order to make alternative simulations of both. The model shows, for example, that when rain increases, the flow of water on the river also increases, increasing suspended solids and sediments and reducing the dissolved oxygen. These factors increase fish mortality, as expected from experts and the literature.

2.5.3 Management and Sustainability

Sustainable development is hampered by limitations on the available knowledge about important interactions and by difficulties to integrate the broad variety of regional problems into typical patterns of global change. Eisenack and Petschel-Held (2002) describe a QSIM model for understanding the interactions between nature and society. Their QR model helps to identify scenarios under which regional land-use changes due to small holders agriculture in developing countries following the 'impoverishment-degradation' spiral. The outcome depends on how the small holders achieve their daily income and how this relates to environmental conditions around them. Eisenack (2003) addresses two threads of the debate on sustainable fisheries: participatory management frameworks and 'ichtyocentric' control strategies. A model of management framework is set up, composed of economic, ecological and political aspects, upon which viability criteria are posed. Then the author investigates how different management strategies change the structure of the resulting state transition graph to conclude that a qualitative viability analysis can be a helpful first step for the design of controllers or the assessment of management frameworks.

2.5.4 Details in Qualitative Algebra

Guerrin (1991; 1992) developed a system (SIMAO) for simulating the interpretation of measurements, observations and analyses, commonly done on

aquatic ecosystems for management purposes. His approach includes directed causal graphs and a qualitative algebra used to combine heterogeneous knowledge obtained by measurements (numerical), observations (linguistic) and calculations. With the support of causal graphs, SIMAO is able to reason with causal relations such as "an increase in photosynthesis decreases the CO2 concentration in water, which in turn (…) hence a risk of decrease of fish production" and to calculate the values of variables using the qualitative algebra. This qualitative algebra was also applied to other biological problems, including photosynthesis (Hunt and Cooke 1994) and the life cycle of a plant population (Salles et al. 1996). Guerrin proposes that this approach could be an option to be used in controlled ecological life support systems (CELSS), modelling, simulation, and control (Guerrin et al. 1994).

2.5.5 Details in Automated Model Building

Applying QR to ecological systems pose new challenges for automatic model building. Rickel and Porter (1997) describe in the domain of plant physiology an approach for answering predictive questions. Depending on the question their approach automatically finds a model with the simplest level of detail adequate for answering that question. A particular feature of their approach is the ability to move between different timescales. Keppens and Shen (2002) address the problem of user preferences in the case of incomplete knowledge. They introduce an order of magnitude preference calculus to handle reasoning with preferences. Their models describe how the Mediterranean vegetation is being affected by various climate related factors, managed and accidental fires, and cattle farming.

2.5.6 Diagnosis

Diagnosis (finding the cause of undesired behaviour) is a promising area for applications for model-based reasoning in ecology (Struss and Heller 1998). Heller and Struss (2002) use model-based technology to support the tasks of situation assessment (determining the actual state of the system) and therapy recognition (determining what can be done to recover from the undesired behaviour). Their work concerns rivers and water treatment plants.

2.6 Conclusion

Representing qualitative ecological knowledge is of great interest for ecological modelling. QR provides means to build conceptual models and to make qualitative knowledge explicit, organized and manageable by means of symbolic computing. This chapter discusses the main characteristics of QR using well-known examples. It also shows how this technology can be used to represent ecological knowledge and an overview is given of ecological applications that have already been developed using QR. Ongoing QR research focuses on improving QR tools and technology. An additional goal is to integrate quantitative knowledge with qualitative knowledge. In a collaborative work with ecologists, particularly in the construction of reusable knowledge libraries, it is possible to foresee a wider range of applications to ecological modelling and better ways of dealing with the complexity of ecological and environmental systems. But most of all, the deployment of QR technology for ecological purposes should become an important goal in itself because, as pointed out by (Rykiel 1989), "many questions of interest in ecology can be answered in terms of 'better or worse', 'more or less', 'sooner or later', etc." and when quantitative methods are inadequate or lacking, it is still possible to make estimates, predictions, and decisions with scientific support.

References

Addanki S, Cremonini R, Penberthy JS (1991) Graphs of models, Artificial Intelligence, 51:1-3, 145-177

Amador F, Finkelstein A, Weld D (1993) Real-time self-explanatory simulation. Proceedings of the 11th International Conference on Artificial Intelligence, AAAI'93, Menlo Park, California, 562-567

Antunes MP, Seixas MJ, Câmara AS, Pinheiro M (1987) A New Method for Qualitative Simulation of Water Resource Systems - 2 Applications. Water Resources Research, 23: 2019-2022

Bessa Machado V, Bredeweg B (2003) Building Qualitative Models with HOMER: A Study in Usability and Support. Proceedings of the 17th International Workshop on Qualitative Reasoning (QR'03), Salles, P. and Bredeweg, B. (Eds.), 39-46, Brasilia, Brazil, August 20-22

Biswas G, Schwartz D, Bransford J (2001) Technology Support for Complex Problem Solving: From SAD Environments to AI. In: Forbus, K. and Feltovich, P. (Eds.) Smart Machines in Education: The coming revolution in educational technology, AAAI Press

Bobrow, D.G. (Ed.) (1984) Qualitative reasoning about physical systems. Elsevier Science, Amsterdam, The Netherlands (reprint from the Journal Artificial Intelligence, volume 24, 1984)

Bossel H (1986) Ecological Systems Analysis: An Introduction to Modelling and Simulation. University of Kassel, Germany, Environmental Research Group, Technical Report

Bouwer A, Bredeweg B (2001) VisiGarp: Graphical Representation of Qualitative Simulation Models. In Moore, J.D., Redfield G.L. and Johnson, J.L. (Eds.), Artificial Intelligence in Education: AI-ED in the Wired and Wireless Future, 294-305, IOS-Press/Ohmsha, Osaka, Japan

Bredeweg B (1992) Expertise in Qualitative Prediction of Behaviour. PhD thesis, University of Amsterdam, Amsterdam

Bredeweg B, Winkels R (1998) Qualitative models in interactive learning environments: an introduction. Interactive Learning Environments, 5: 1-2, 1-18

Brown JS, Burton RR, Kleer J de (1982) Pedagogical, natural language and knowledge engineering techniques in SOPHIE I, II and III. In Sleeman, D. and Brown, J.S. (Eds.), Intelligent Tutoring Systems, pages 227-282, Academic Press, New York USA

Câmara AS, Antunes PC, Pinheiro MD, Seixas MJ (1987) Linguistic dynamic simulation - A new approach. Simulation, 49: 5, 208-212

Collins J, Forbus K (1989) Building qualitative models of thermodynamic processes. ILS Technical report, Computer Science Department, Northwestern University, Evanston, USA

Eisenack K, Petschel-Held G (2002) Graph Theoretical Analysis of Qualitative Models in Sustainability Science. Proceedings of the International Workshop on Qualitative Reasoning, (QR'02), Agell, N. and Ortega, J.A. (Eds.), 53-60, Sitges/Barcelona, Spain, June 10-12, 2002

Eisenack K (2003) Qualitative Viablility Analysis of a Bio-Socio-Economic System. Proceedings of the17th International Workshop on Qualitative Reasoning, (QR'03), Salles, P. and Bredeweg, B. (Eds.), 63-70, Brasília, Brazil, August 20-22, 2003

Falkenhainer B, Forbus K (1991) Compositional modeling: Finding the right model for the Job. Artificial Intelligence, 51: 1-3, 95-143

Forbus KD (1984) Qualitative process theory. Artificial Intelligence, 24: 1-3, 85-168

Forbus KD (1986) The Qualitative Process Engine. University of Illinois, Department of Computer Science Technical Report. Reprinted in Weld, D.S. and Kleer, J. de (Eds.) (1990) Readings in Qualitative Reasoning about Physical Systems, Morgan Kaufmann, San Mateo, California, USA

Forbus KD (1988) Qualitative physics: past, present and future. In: H.E. Shrobe (Ed.), Exploring artificial intelligence, 239-296, Morgan Kaufmann, San Mateo. USA

Forbus KD, Whalley P, Everett J, Ureel L, Brokowski M, Baher J, Kuehne S (1999) CyclePad: An articulate virtual laboratory for engineering thermodynamics, Artificial Intelligence, 114: 1-2, 297-347

Forbus KD, Carney K, Harris R, Sherin BL (2001) A qualitative modeling environment for middle-school students: A progress report. In: Biswas, G. (Ed.), Proceedings of the 15th International Workshop on Qualitative Reasoning (QR'01), 65-72, St. Mary's University, San Antonio, Texas

Fromherz MPJ, Bobrow DG, Kleer J de (2003) Model-based Computing For Design and Control of Reconfigurable Systems, AI Magazine, 24: 4, 102-130

Guerrin F (1991) Qualitative Reasoning about an Ecological Process: Interpretation in Hydroecology. Ecological Modelling, 59: 165-201

Guerrin F (1992) Model-based Interpretation of Measurements, Analysis and Observations of an Ecological Process. AI Applications, 6: 3, 89-101

Guerrin F, Bousson K, Steyer JPh, Travé-Massuyès L (1994) Qualitative Reasoning Methods for CELSS Modeling. Advances in Space Research, 14: 11, 307-312

Guerrin F, Dumas J (2001a) Knowledge representation and qualitative simulation of salmon reed functioning. Part I: qualitative modelling and simulation. BioSystems, 59: 75-84

Guerrin F, Dumas J (2001b) Knowledge representation and qualitative simulation of salmon reed functioning. Part II: qualitative model of reeds. BioSystems, 59: 85-108

Heller U, Struss P (2002) Consistency-based Problem Solving for Environmental Decision Support. Computer - Aided Civil and Infrastructure Engineering, 17: 79-92

Hollan JD, Hutchins EL, Weitzman L (1984) STEAMER: An interactive inspectable, simulation based training systems. AI Magazine, 5: 15-27

Hunt JE, Cooke DE (1994). Qualitative modeling photosynthesis. Applied Artificial Intelligence, 8: 307-332

Iwasaki Y, Simon HA (1986) Causality in device behaviour. Artificial Intelligence, 29: 3-32

Keppens J, Shen Q (2002) On Supporting Dynamic Constraint Satisfaction with Order of Magnitude Preferences. Proceedings of the 16th International workshop on Qualitative Reasoning, (QR'02), Agell, N. and Ortega, J.A. (Eds.), 143-150, Sitges/Barcelona, Spain, June 10-12, 2002

Kim H (1993) Qualitative reasoning about fluids and mechanics. Ph.D. thesis, Computer Science Department, Northwestern University, Evanston, USA

Kleer J de, Brown JS (1984) A qualitative physics based on confluences. Artificial Intelligence, 24: 1-3, 7-83

Kleer J de, Brown JS (1986) Theories of causal ordering. Artificial Intelligence, 29: 33-61

Kuipers B (1986) Qualitative simulation. Artificial Intelligence, 29: 289-388

Kuipers B (1994) Qualitative reasoning: modeling and simulation with incomplete knowledge. MIT Press, Cambridge, Massachusetts

May RM (1973) Qualitative stability in model ecosystems. Ecology, 54: 3, 638-641

McIntosh BS (2003) Qualitative modelling with imprecise ecological knowledge: a framework for simulation. Environmental Modelling and Software, 18: 295-307

Moore AD, Noble IR (1990) An Individualistic Model of Vegetation Stand Dynamics. Journal of Environment Management, 31: 61-81

Moore AD, Noble IR (1993) Automatic Model Simplification: the Generation of Replacement Sequences and their Use in Vegetation Modelling. Ecological Modelling, 70: 137-157

Neumann M, Bredeweg B (2004) A qualitative model of the nutrient spiraling in lotic ecosystems to support decision makers for river management. Proceedings of the 18th International Workshop on Qualitative Reasoning (QR'04), de Kleer, J. and Forbus, K.D. (Eds.), 159-164, Evanston, USA, August 2-4, 2004

Noble IR, Slatyer RO (1980) The Use of Vital Attributes to Predict Successional Changes in Plant Communities Subject to Recurrent Disturbances. Vegetatio 43: 5-21

Nuttle T, Bredeweg B, Salles P (2004) Qualitative Reasoning about Food Webs: Exploring Alternative Representations. Proceedings of the 18th International Workshop on Qualitative Reasoning (QR'04), de Kleer, J. and Forbus, K.D. (Eds.), 89-96, Evanston, USA, August 2-4, 2004

Price C, Struss P (2003) Model-based Systems in the automotive Industry, AI Magazine, 24: 4, 17-43

Raiman O (1986) Order of magnitude reasoning. Proceedings of the AAAI, 100-104, San Mateo, California, Morgan Kaufmann

Rickel J, Porter B (1997) Automated modeling of complex systems to answer prediction questions. Artificial Intelligence, 93: 201-260

Rykiel EJ (1989) Artificial Intelligence and Expert Systems in Ecology and Natural Resource Management. Ecological Modelling, 46: 3-8

Salles P (1997) Qualitative models in ecology and their use in learning environments. Ph.D. thesis, University of Edinburgh, Edinburgh, Scotland, UK

Salles P, Bredeweg B (1997) Building Qualitative Models in Ecology. In: Ironi, L. (Ed.) Proceedings of the 11th International Workshop on Qualitative Reasoning (QR'97). Instituto di Analisi Numerica C.N.R., Pubblicazioni no. 1036 , Pavia, Italy

Salles P, Bredeweg B (2003) Qualitative Reasoning about Population and Community Ecology. AI Magazine, 24: 4, 77-90

Salles P, Bredeweg B (2005) Modelling Population and Community Dynamics with Qualitative Reasoning. Ecological Modelling, in press

Salles P, Bredeweg B, Araujo S, Neto W (2003a) Qualitative models of interactions between two populations. AI Communications, 16: 4, 291-308

Salles P, Bredeweg B, Bensusan N (2003b). The Ant's Garden: Qualitative Models of Complex Interactions between Populations. Proceedings of the 17th International Workshop on Qualitative Reasoning (QR'03), Salles, P. and Bredeweg, B. (Eds.), 163-170, Brasilia, Brazil, August 20-22, 2003

Salles P, Bredeweg B, Araujo S (2003c). Qualitative Models of Stream Ecosystem Recovery: Exploratory Studies. Proceedings of the 17th International Workshop on Qualitative Reasoning (QR'03), Salles, P. and Bredeweg, B. (Eds.), pages 155-162, Brasilia, Brazil, August 20-22

Salles P, Muetzelfeldt RI, Pain H (1996) Qualitative Models in Ecology and their Use in Intelligent Tutoring Systems. Proceedings of 10th International Workshop on Qualitative Reasoning (QR'96), Iwasaki, Y. and Farquhar, A. (Eds.). AAAI Technical Report WS-96-01

Struss P, Heller U (1998) Process-oriented Modelling and Diagnosis - Revising and Extending the Theory of Diagnosis from First Principles. Working Notes of the Ninth International Workshop on Principles on Diagnosis (DX-98), Sea Crest, Cape Cod, USA

Tullos DD, Neumann M, Sanchez JJA (2004) Development of a Qualitative Model for Investigating Benthic Community Response to Anthropogenic Activities. Proceedings of the 18th International Workshop on Qualitative Reasoning (QR'04), de Kleer J. and Forbus, K.D. (Eds.), pages 179-185, Evanston, USA, August 2-4, 2004

Weld DS (1988) Comparative analysis. Artificial Intelligence, 36: 333-374

Weld DS, Kleer J de (Eds.) (1990) Readings in Qualitative Reasoning about Physical Systems, Morgan Kaufmann, San Mateo, California, USA.

Williams BC, Ingham M, Chung S, Elliott P, Hofbaur M (2003) Model-based Programming of Fault-Aware Systems, AI Magazine, 24: 4, 61-76

Chapter 3

Ecological Applications of Non-supervised Artificial Neural Networks

J.L. Giraudel · S. Lek

3.1 Introduction

For ecological data, cluster analysis (CA) is basically a classification technique for sorting sample units into groups based upon their resemblance. Several algorithms permit similar work, but due to the heuristic nature of the methods, it is impossible to choose the 'best' one (Van Tongeren 1995). Some conventional methods need prior knowledge of the number or of the size of the clusters. On the other hand, particular shapes of the clusters may lead to erroneous conclusions. The results are commonly displayed on dendrograms becoming hard to interpret for huge datasets.

Inspired by the structure and the mechanism of the human brain, the Artificial Neural Networks (ANNs) should be a convenient alternative tool to traditional statistical methods. ANNs have already been successfully used in ecology (Lek and Guegan 1999). Whereas, the backpropagation algorithm of ANNs in a supervised learning way is mostly used in the ecological applications (Lek et al. 2000), only a few applications of ANNs with non-supervised learning also called Self-Organizing Maps (SOM) are known. Nevertheless, SOM has been demonstrated in patternizing ecological communities (Chon et al. 1996) and has been applied to the analysis of community data (Foody 1999) or to model microsatellite data (Giraudel et al. 2000).

The Kohonen SOM is the prototype of non-supervised ANNs invented by Kohonen (1995). It performs a topology-preserving projection of the data space onto a regular two-dimensional space and can be used to visualize clusters effectively.

The main aim of this paper is to demonstrate a practical methodology for the best use of SOMs for community classification. The presentation will be made using a well-known dataset: upland forests in Wisconsin, USA (Peet and Loucks 1977). Firstly, the methodology leading to a good learning process will be specified. It will be shown that the SOM is a good tool to interpret the classification with abundance data or abiotic variables. Then, computing the unified-matrix, an effective way to obtain clusters will be explained and some indices to evaluate the quality of the map will be given.

3.2
How to Compute a Self-Organizing Map (SOM) with an Abundance Dataset?

3.2.1
A Dataset for Demonstrations

We consider a classical abundance dataset in the form of a matrix with n rows and p columns, the rows representing the species and the columns the sample units (SUs); in addition, for each SU, environmental factors are sometimes available (Table 2.1). Then, each SU can be considered as a vector in the n-dimensional space R^n. Measurements of abundance are density, presence, frequency, biomass.

The SOM algorithm has been applied to a classical dataset (Table 2.2): the distribution of 8 tree species (n=8) in 10 sites (p=10) in Southern Wisconsin - USA (Peet and Loucks 1977). This data matrix has been expanded to provide information for soil texture (five classes of percentage in the A1 horizon). This dataset has been particularly used by Ludwig and Reynolds (1988) who used it with some classical methods of ordination and of classification. It has already been used by Chon et al. (1996) to check the feasibility of the Kohonen algorithm in clustering ecological data providing a dendrogram based on the method of average linkage between groups (Fig. 3.1). Although this dataset is relatively small and simple, it is convenient for demonstrating the methods used in this paper.

Table 3.1. An ecological data matrix, n species ($sp_1,\ldots,sp_n$) are observed in p sample units ($SU_1,\ldots,SU_p$). x_{ij} can be species abondance or species proportion or presence-absence or preprocessing data.

		sample units			
		SU_1	SU_2	…	SU_p
	sp_1	x_{11}	x_{12}	…	x_{1p}
	sp_2	x_{21}	x_{22}	…	x_{2p}
species	.	.	.		.
	sp_n	x_{n1}	x_{n2}	…	x_{np}
Site factor	f	y_1	y_2	…	y_p

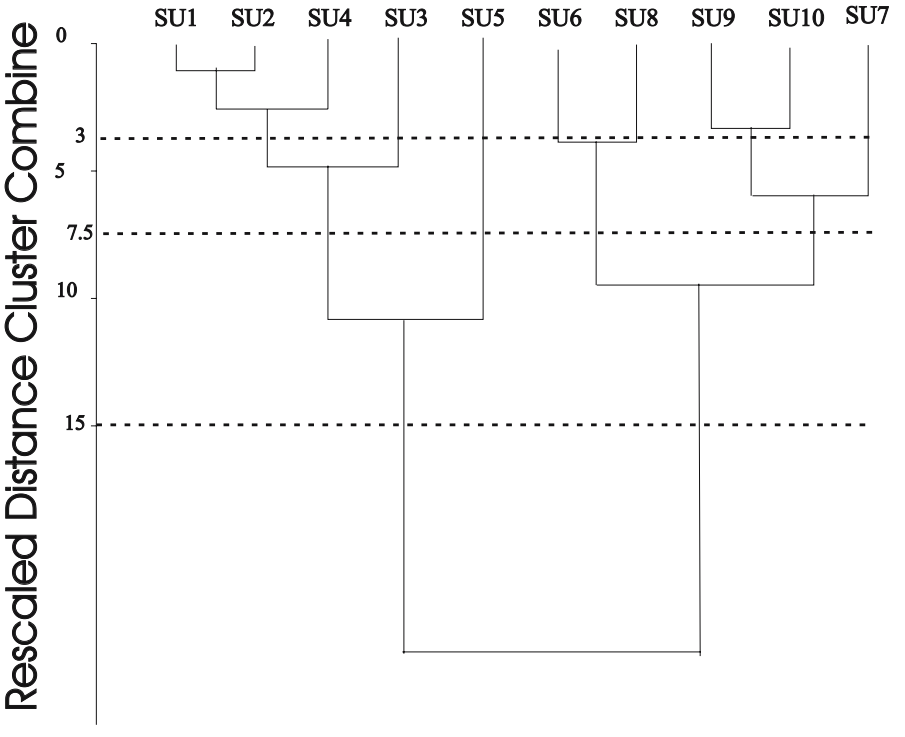


Figure 3.1. Dendrogram for clustering the ten upland forest communities: data from Ludwig and Reynolds (1988), dendrogram from Chon et al. (1996)

Table 3.2. Data matrix of abundances for eight trees in 10 upland forest sampling units, southern Wisconsin Peet and Loucks (1977)

Species		Sample Units									
Name	Number	SU1	SU2	SU3	SU4	SU5	SU6	SU7	SU8	SU9	SU10
Bur oak	(1)	9	8	3	5	6	0	5	0	0	0
Black oak	(2)	8	9	8	7	0	0	0	0	0	0
White oak	(3)	5	4	9	9	7	7	4	6	0	2
Red oak	(4)	3	4	0	6	9	8	7	6	4	3
American elm	(5)	2	2	4	5	6	0	5	0	2	5
Basswood	(6)	0	0	0	0	2	7	6	6	7	6
Ironwood	(7)	0	0	0	0	0	0	7	4	6	5
Sugar maple	(8)	0	0	0	0	0	5	4	8	8	9
Soil texture		4	5	3	2	1	1	2	1	1	1

3.2.2
The Self-Organizing Map (SOM) Algorithm

This section explains how the SOM algorithm can be adapted to an abundance dataset. The SOM algorithm has been described by Kohonen (1982) in the early eighties. Since that time, it has been most widely used for data mining and knowledge discovery.

The SOM algorithm performs a non-linear projection of the dataset onto a rectangular grid (r rows and c columns) laid out on a hexagonal lattice with S hexagons ($S = r.c$): the Kohonen map. Formally, the Kohonen neural network consists of two layers: the first one (input layer) is connected to each vector of the dataset, the second one (output layer) forms a two-dimensional array of nodes (Fig. 3.2). The main characteristic of the SOM classification is the conservation of the topology. Close sample units (stands or stations) are associated with the same node or to nearby nodes on the map.

For this purpose, in each hexagon, a virtual unit (VU) will be considered (Figure 3.3). The VUs $(VU_k)_{1\le k\le S}$ are in fact, virtual sites with species abundance (w_{ik}) to be computed (Table 3.3).

Table 3.3. Components of the virtual units in the output layer

	virtual units			
	VU_1	VU_2	…	VU_S
	sp_1 $w_{11}(t)$	$w_{12}(t)$	…	$w_{1S}(t)$
	sp_2 $w_{1S}(t)$	$w_{22}(t)$	…	$w_{2S}(t)$
species	. .	.		.
	sp_n $w_{n1}(t)$	$w_{n2}(t)$	…	$w_{nS}(t)$

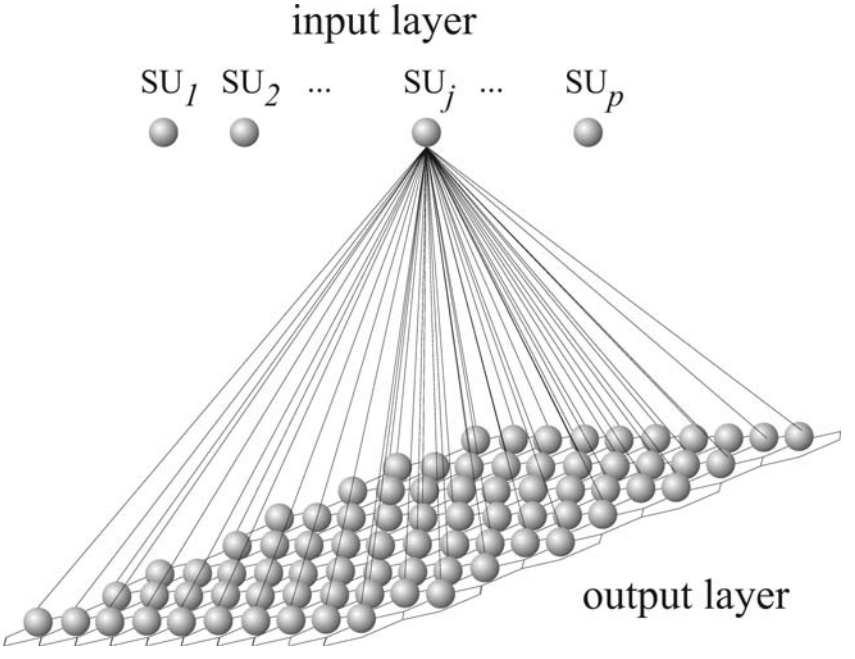


Figure 3.2. A two-dimensional self-organizing map. Each sphere symbolizes each neuron at the input layer (data row) and the output layer (Kohonen map).

The modifications of the VUs are made through an ANN. Imitating the organization of the human brain, the ANN has a learning ability: the components (w_{ik}) of each virtual unit (actually, species abundance) are computed during a training phase. The modifications of each (w_{ik}) take place by iterative adjustments based on the species abundance of the sample units presented in the input layer. As opposed to a supervised learning process, for each input unit, the desired output is unknown, we are referring to unsupervised learning. The aim of the training process is that the distribution of the VUs on the map should reflect the distribution of the SUs. Once the training phase is completed, the VUs are left unchanged.

The learning steps are well known (Kohonen 1995) and can be summarized as follows:

Step 1: Epoch t=0, the virtual units $\left(VU_k\right)_{1\le k\le S}$ are initialized with random samples drawn from the input dataset.

Step 2: A sample unit SU_j is randomly chosen as an input unit.

Step 3: The distance between SU_j and every virtual unit is computed.

Step 4: The virtual unit VU_c closest to input SU_j is chosen as the winning neuron. VU_c is called the Best Matching Unit (BMU).

Step 5: The virtual units $\left(VU_k\right)_{1\le k\le S}$ are updated with the rule:

$$w_{ik}(t+1)=w_{ik}(t)+h_{ck}(t)\left[x_{ij}(t)-w_{ik}(t)\right] \quad (3.1)$$

Step 6: Increase time t to $t + 1$. If $t < t_{max}$ then go to step 2 else stop the training.

There is no precise rule for the choice of the size of the grid, it can be chosen larger than the number of SUs. A hexagonal lattice has to be preferred, because it does not favor horizontal or vertical directions as much as the rectangular array (Kohonen 1995). For our dataset, the SOM has been formed with 16 hexagons: 4 rows and 4 columns ($c = 4$; $r = 4$; $S = 16$) .

In step 5, in the equation (3.1), the function $h_{ci}(t)$ is called the neighborhood function and plays a very central role. Several choices can be made for the definition of the neighborhood function. If there are less than a few hundred nodes, selection of neighborhood functions is not very crucial, however, special caution is required in the choice of the size of the neighborhood affected by the learning (see Kohonen 1995). The simplest neighborhood function is the bubble: it is constant in the neighborhood of the BMU and zero elsewhere. In this work, we have chosen a neighborhood written in terms of the Gaussian function:

$$h_{ck}(t)=\alpha(t).\exp\left(-\frac{\left\|r_k-r_c\right\|^2}{2\sigma^2(t)}\right) \quad (3.2)$$

$\left\|r_k-r_c\right\|^2$ is the Euclidean distance on the map between the BMU VU_c and each virtual unit VU_k.

δ is a decreasing function of the time which defines the width of the part of the map affected by the learning process.

α is the ``learning-rate factor'', it is a decreasing function of the time.

α and δ both converge towards 0.

The learning process is broken down into two parts:

The ordering phase: during this phase, the virtual stations are highly modified in a wide neighborhood of the Best Matching Unit. So, this occurs with large values for α and δ.

The tuning phase: when this second phase takes place, only the virtual units adjacent to the BMU are modified. This phase is much longer than the former one and α is decreasing very slowly towards 0.

According to Kohonen's advice, the number of steps must be at least 500 times the number of network units of which around 2 000 steps are for the ordering phase. For the Wisconsin forest community data, the learning phase has been broken down into 2 000 steps for the ordering phase and 80 000 steps for the tuning phase.

The species abundance can be preprocessed before the learning process of the SOM. There is no limitation: transformations such as logarithmic or

Table 4.4. Species proportions in each virtual unit. These proportions are computed during the learning process of the SOM using the Whittaker's relative transformation.

Virtual Units	Species							
	sp1	sp2	sp3	sp4	sp5	sp6	sp7	sp8
VU1	0.13	0.33	0.38	0.00	0.17	0.00	0.00	0.00
VU2	0.14	0.28	0.33	0.09	0.16	0.00	0.00	0.00
VU3	0.07	0.10	0.27	0.24	0.07	0.14	0.00	0.10
VU4	0.00	0.00	0.24	0.26	0.00	0.24	0.04	0.21
VU5	0.23	0.30	0.24	0.11	0.12	0.00	0.00	0.00
VU6	0.16	0.22	0.28	0.19	0.16	0.00	0.00	0.00
VU7	0.05	0.07	0.25	0.23	0.05	0.16	0.04	0.15
VU8	0.00	0.00	0.22	0.23	0.00	0.22	0.09	0.24
VU9	0.31	0.31	0.17	0.13	0.07	0.00	0.00	0.00
VU10	0.25	0.21	0.21	0.19	0.13	0.02	0.00	0.00
VU11	0.16	0.07	0.21	0.23	0.16	0.08	0.06	0.03
VU12	0.03	0.00	0.10	0.16	0.09	0.20	0.18	0.24
VU13	0.27	0.21	0.19	0.19	0.12	0.02	0.00	0.00
VU14	0.18	0.00	0.19	0.26	0.18	0.10	0.06	0.03
VU15	0.10	0.00	0.11	0.19	0.14	0.17	0.15	0.16
VU16	0.03	0.00	0.05	0.14	0.12	0.21	0.19	0.26

standardization may be applied. Moreover, in step 3, as opposed to what happens with the most used clustering methods, the Euclidean distance is not the only possibility and some measurements of ecological similarity can be chosen. However, in this case, the use of the learning rule (eq. 3.1) has to be adapted in order to be compatible with the chosen distance (Kaski 1997).

For instance in this work, as suggested by Orloci (1978), the Whittaker's relative transformation (1952) for absolute distance has been used. In this way, the distance between 2 units SU_i and SU_j respectively defined with species abundance is computed by: (x_{1i} , …, x_{ni}) and (x_{1j} , …, x_{nj}

$$D(SU_i,SU_j)=\sum_{i=1}^{n}\left|\frac{x_{li}}{\sum_{l=1}^{n}x_{li}}-\frac{x_{lj}}{\sum_{l=1}^{n}x_{lj}}\right| \tag{3.3}$$

And the learning rule (eq. 3.1) becomes:

$$w_{ik}(t+1)=\frac{w_{ik}(t)+h_{ck}(t)\left[x_{ij}(t)-w_{ik}(t)\right]}{\sum_{l=1}^{n}\left(w_{lk}(t)+h_{ck}(t)\left[x_{lj}(t)-w_{lk}(t)\right]\right)} \tag{3.4}$$

3.3 How to Use a Self-Organizing Map with an Abundance Dataset?

3.3.1 Mapping the Stations

The SOM has been computed in order that the VU distribution should follow the SU distribution. Most usually, the SOM is used to show SUs in the corresponding hexagons, SUs are mapped by a nearest neighbor method. For this purpose, the BMU is computed for each SU and this SU is put in the corresponding hexagon. The results obtained with the Euclidean distance are given in Fig. 3.4 and those obtained with the Whittaker's relative transformation are given in Fig. 3.5.

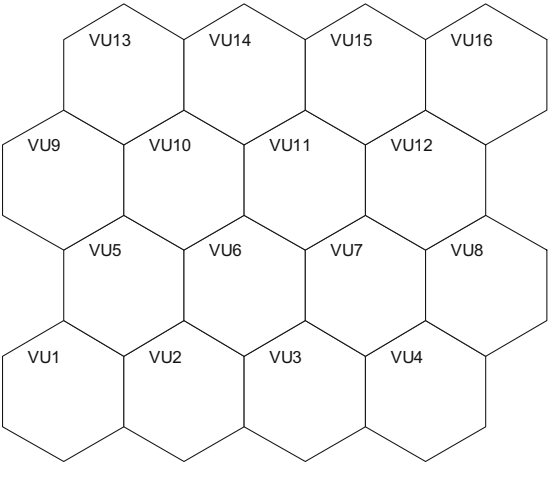


Figure 3.3. The output layer of the Kohonen network. A virtual unit VU_k has been defined in each hexagon of the rectangular grid.

In order to observe how the choice of the measurement methods could be reflected in community groupings, an other map has been built using Euclidean distance.

When the learning process is finished, a map with *S* hexagons is obtained and in each hexagon, there is a virtual station in which species abundance has been computed. For the upland forest data, the species abundance of the VUs after the learning process using the Whittaker's relative transformation can be seen in Table 3.4.

Then this map can be used in different ways: representation of the stations on the map, component planes (the species composition of each VU can be used to display the distribution of each species), representation of an abiotic variable on the map, determination of clusters in the SU space. Moreover, some new data unknown during the learning process can be added to the map. It is very convenient to compute the BMUs for these data and to map them.

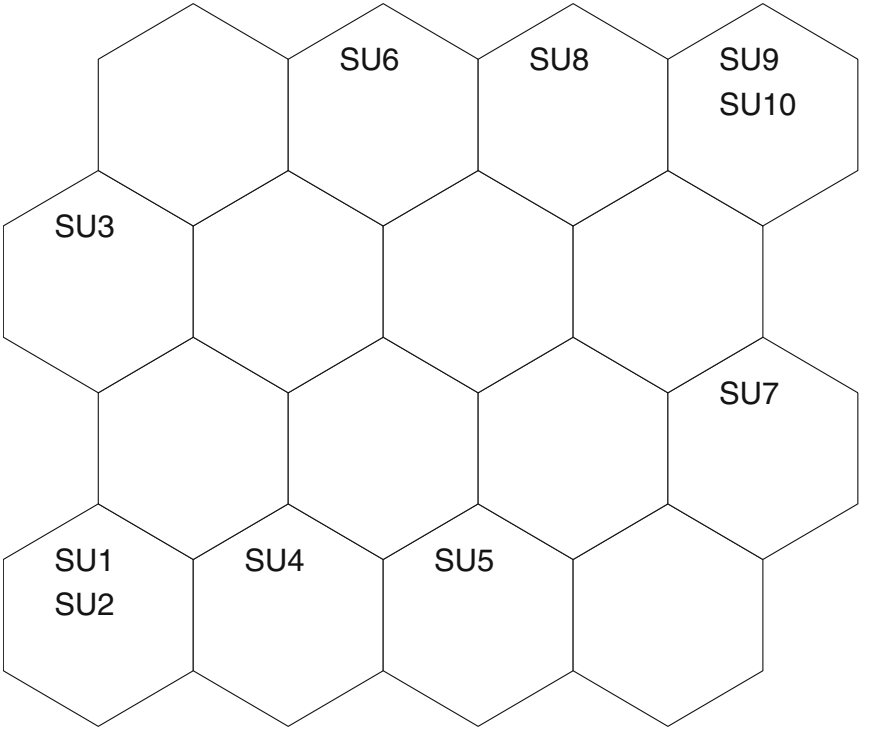


Figure 3.4. 10 upland forest sites mapped on the Self-Organizing Map using the Euclidean distance.

However, the ordination on the map of these data will be consistent only if these items can be assumed to follow the same distribution as the items that were taken into account during the learning process (Kaski 1997).

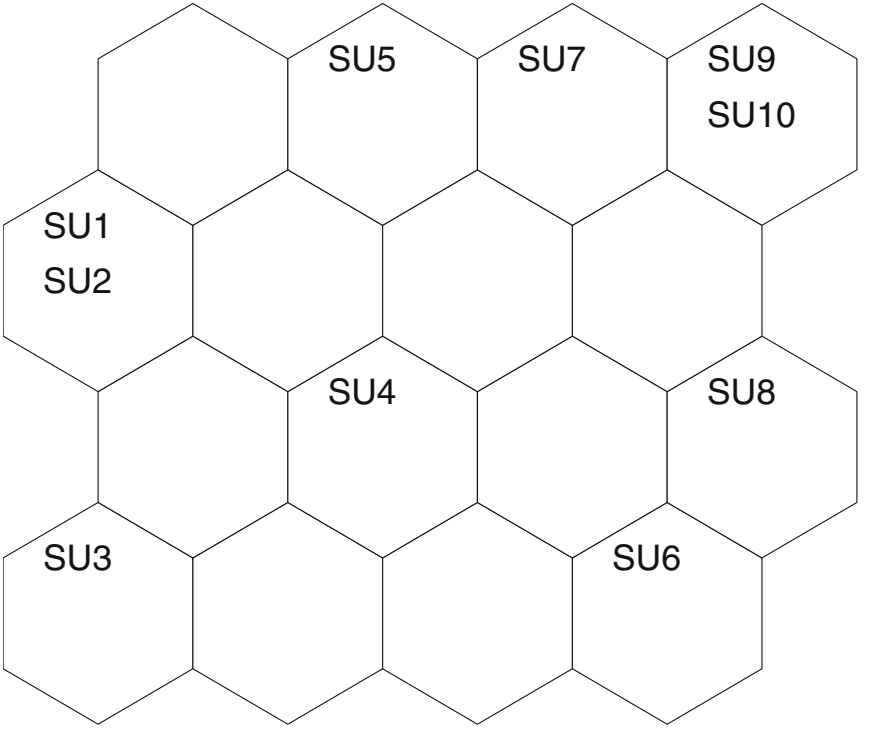


Figure 3.5. 10 upland forest sites mapped on the Self-Organizing Map using the Whittaker's relative transformation.

3.3.2 Displaying a Variable

Component plane representation visualizes the species abundance of the VUs. This representation can be considered as a "sliced" version of the SOM. Each plane displays each species abundance in the VUs.

For instance, this method has been applied to the upland forest data (using the Whittaker's relative transformation) and the abundance of sugar maple is shown in Fig. 3.6. A grey shade has been used: light colour for poor abundance and dark colour for greater abundance. With this representation, the SOM becomes a powerful tool to analyse the community structure: a large abundance can be seen in the right part of the map (in SUs 6 to 10) and an absence is noted in the left part (SUs 1 to 5).

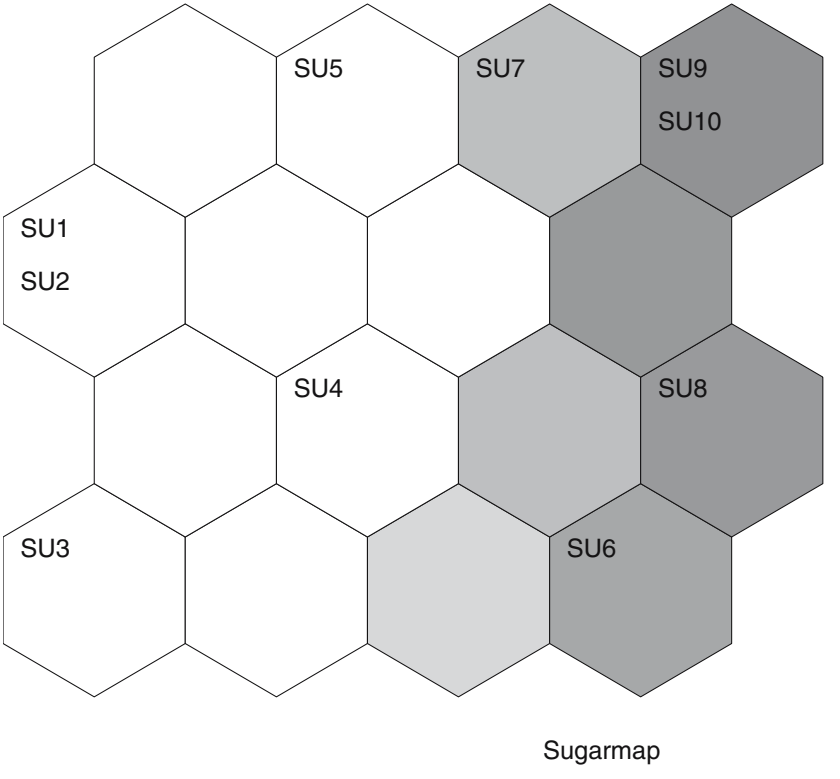


Figure 3.6. Component plane for sugar maple. The gray levels describe the proportion of sugar maple at each location on the map. The map has been computed using the Whittaker's relative transformation

3.3.3 Displaying an Abiotic Variable

The way to display an environmental factor on a SOM is rather easy. The aim is to attribute a value for this factor in each hexagon of the map. For an output unit k (a hexagon), if at least one SU is in the hexagon, we affect the arithmetic mean of the SU factors to the virtual unit VU_k. If no SU is in the hexagon k, the arithmetic mean of the values of the abiotic variable in the adjacent hexagons is chosen. Then, each hexagon is coloured in different levels of grey according to the value of the environmental factor. In Fig. 3.7, the soil texture has been displayed for the upland forest data and a decreasing gradient of this factor can be seen from the left to the right part of the map.

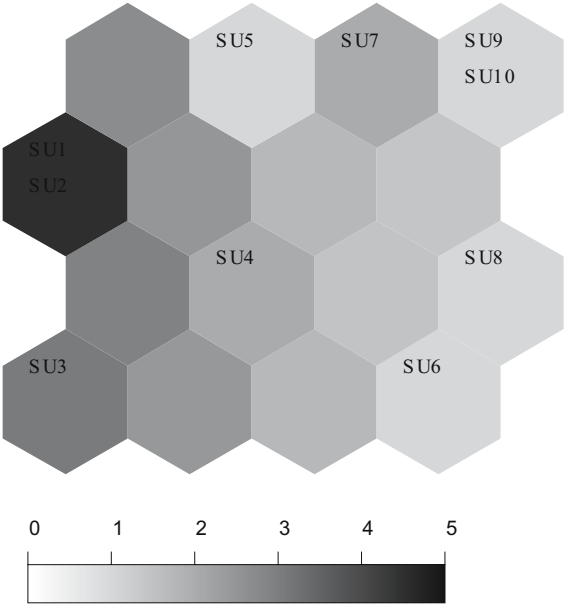


Figure 3.7. Soil texture for the upland forest dataset.

3.3.4
Clustering with a SOM

Visual inspection of the SOM allows some groups to be seen immediately: the SUs in the same hexagon are reputed to be in the same cluster. By this way, 8 clusters are defined for the upland forest dataset (Figs. 3.5, 5): A_{I} (SUs 1, 2), A_{II} (SU 3), A_{III} (SU 4), A_{IV} (SU 5), A_{V} (SU 6), A_{VI} (SU 7), A_{VII} (SU 8), and A_{VIII} (SUs 9 and 10). It can be noticed that these clusters are the same with the Euclidean distance and the Whittaker's relative transformation.

By combining some hexagons, it is possible to form bigger clusters. But, a deficiency of the initial SOM algorithm was the difficulty in detecting the cluster boundaries on the map for units in different hexagons. A few enhancement techniques have been proposed to tackle this problem, for instance, hierarchical feature maps (Miikkulainen 1990) or adaptive coordinates (Merkl and Rauber 1997). However, these two methods are not considered further in this paper, the unified-matrix (U-matrix) approach (Ultsch and Siemon 1990) will be preferred for its ability to provide a very flexible tool for ecologists.

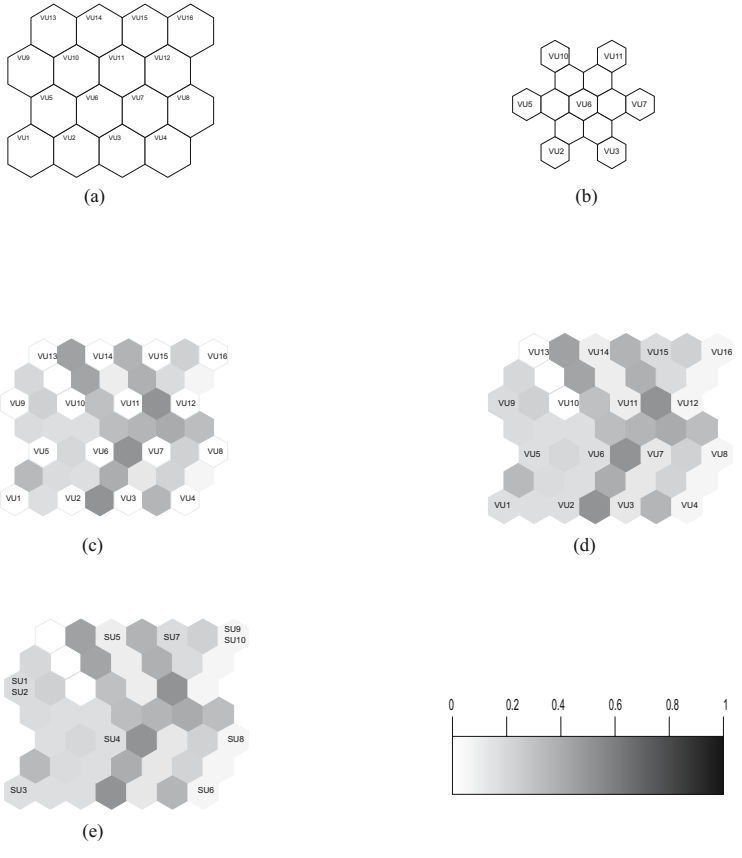
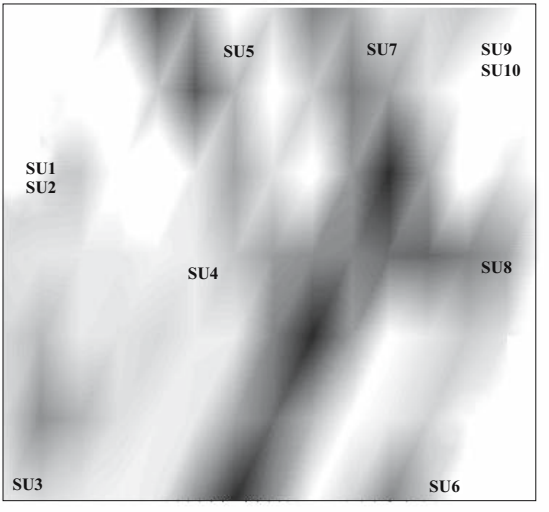


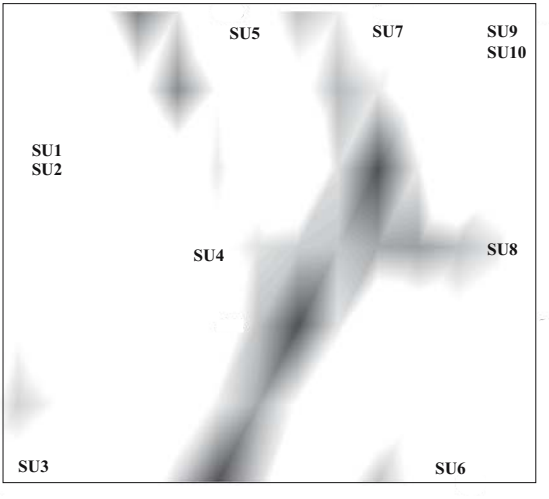
Figure 3.8. A U-matrix representation of the self-organizing map computed using the Whittaker's relative transformation. (a) Location of the virtual units VU_k on the self-organizing map. (b) New hexagons are inserted between adjacent hexagons. (c) Gray levels showing the distance between two adjacent hexagons. Light for short distances and dark for large distances. (d) Each hexagon with virtual units is colored according to the minimum of its adjacent hexagons. (e) Sample units mapped on the U-matrix. Plains (light areas) can be seen separated by ravines (dark areas)

The U-Matrix Display

This method will be presented using the results obtained with the Whittaker's relative transformation. With the U-matrix method, in order to detect a clustering structure, a new map will be built. When the learning process has been completed, a map with c columns and c rows is obtained. In each hexagon of the map, VUs have been defined (Fig. 3.8a) and the species abundances of each VU are known. In order to visualize the cluster structure of the map, the key idea of the



(a)



(b)

Figure 3.9. A U-matrix representation of the self-organizing map computed using the Whittaker's relative transformation. Two different levels of brightness are shown in a) and b)

The U-matrix method is to compute the distance between two VUs located in two adjacent hexagons. High value distances will be used as an indication of cluster boundaries. To visualize the distances, new hexagons will be used and from the

initial grid a new one can be constructed, inserting a new hexagon between each adjacent hexagon (Fig. 3.8 b). So, if c columns and r rows make up the initial grid of the output layer, the U-matrix is a matrix with ($2c$ - 1) columns and ($2r$ - *1*) rows where grey levels show the distance values (Fig. 3.8 c). But distance values are only available for the new hexagons so the unified distance matrix has to be completed. For this purpose, in each hexagon including a virtual unit, a distance has been added, calculated as the minimum of its adjacent hexagons (Fig. 3.8 d). If dark colours are used for large distances and light colours for short distances, the U-matrix can be seen as a landscape displaying the distances between the VUs. This landscape is formed with light plains separated by dark ravines. When, SUs are mapped, the units in the plains are close to each other in the input layer so these SUs are similar (for species abundance) and clusters becomes apparent (Fig. 3.8 e).

An enhancement of this representation can be made: a triangle-based cubic interpolation (Watson 1992) has been applied to the U-matrix. Then, a smooth surface is obtained and displayed with the possibility of changing the brightness of the figure. The brighter the display, the lower the number of clusters becoming apparent.

The outcome of this process for the Wisconsin forests can be seen in Fig. 3.9. In this way, on Fig. 3.9 a, 4 clusters can be made out: B_{I} (SUs 1, 2, 3 and 4), B_{II} (SU 5), B_{III} (SUs 7, 9 and 10), and B_{IV} (SUs 6 and 8). Then on Fig. 3.9b, the SUs can be grouped in 2 clusters C_{I} (SUs 1, 2, 3, 4 and 5) and C_{II} (SUs 6, 7, 8, 9 and 10).

If we consider the U-matix computed with the Euclidean distance (Fig. 3.10), 3 clusters can be made out: E_{I} (SUs 1, 2, 3, 4 and 5), E_{II} (SU 7) and E_{III} (SUs 6, 8, 9 and 10).

So the clustering method with the SOM can be summarized as follows:

1. Training the SOM: the species abundance is computed for each VU.
2. Computing the U-matrix.
3. Mapping the SUs onto the U-matrix.
4. Making the clustering structure apparent for the human expert of the dataset by selecting the brightness of the display.

3.4 Discussion

As already mentioned by Chon et al. (1996), high similarity between the SOM results and the dendrogram (Fig. 3.1) may be observed. The 8 clusters (A_{I} to A_{VIII}) previously defined with the SOM are those identified on the dendrogram by a dotted line at distance 3, except for the SU 4 which is grouped with the SUs 1 and 2 on the dendrogram. The U-matrix enhances the SOM and brings more accurate

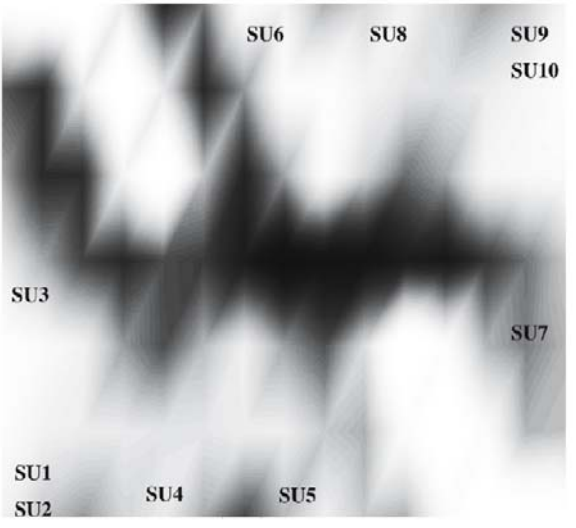


Figure 3.10. A U-matrix representation of the self-organizing map computed using the Euclidean distance.

results. The 4 clusters (B_{I} to B_{IV}) seen in the Figure 3.9a are exactly those identified on the dendrogram by a dotted line at distance 7.5. In the same way, with the 2 clusters C_{I} and C_{II} in the SOM (Fig. 3.9b) at distance 15 on the dendrogram.

The clusters defined with the SOM built using the Euclidean distance are also very similar with those obtained with the dendrogram but the cluster E_{II} including the SU 7. This new cluster can be explained by the high value of the species abundances in the SU 7, the Euclidean distance puts greater importance on the absolute quantities of species and less importance on their relative proportions.

These results constitute a validation of the use of SOMs associated with a U-matrix for clustering ecological data. With a large dataset, when dendrograms become very difficult to read, the SOM and the U-matrix are able to provide a very convenient visualization. These methods have been applied on a large dataset: 250 sampling sites were classified according to the similarity of their invertebrate species composition (with 283 species) (Céréghino et al. 2001). But it is worth noticing that the U-matrix is not a "ready made" clustering algorithm but rather a tool for the inspection of high dimensional data (Ultsch and Siemon 1990). The clusters have to be ``seen" on the map by the human dataset expert. In this way, the expert can define all types of clusters including the non-convex ones.

The U-matrix display is in fact 3-dimensional. Nowadays, software allows a 3D representation in which interactive rotations can be carried out. Some applications

of such techniques have been proposed with SOMs (Vesanto et al. 1998) and could be turned to good account with large ecological datasets.

With unsupervised training, the map quality cannot easily be estimated: the SOM algorithm is not based on the minimization of a goal function. However, several criteria have been suggested, for instance, average quantization error and topological quantization error may be used to quantify topology preservation (Hämäläinen 1994, Kraaijveld et al. 1995). For ecological data, the Euclidean distance is not necessarily the only possibility, thus the use of the topographic error ε. (Kiviluoto 1996) can be suggested. The topographic error ε gives the proportion of sample units for which the first BMU and the second BMU are not in adjacent hexagons on the map. On the forest upland dataset, ε has been computed equal to 0 - for all SUs, the first BMU and the second BMU are in two adjacent hexagons - this is the proof of an excellent learning process and the achievement of a very smooth map.

Sometimes, in a few sites, some species abundances are not known. In such a case, conventional clustering methods cannot be used. However, a SOM can be computed in the following way: in step 3, if some components of SU_j are missing, the computation of the distances between SU_j and each virtual unit has to be made only with the available components. The BMU is worked out and updated with its neighbors (eq. 3.1) using only the available components of SU_j. If only a small proportion of the components of the data vector is missing, better results are obtained in this way than by discarding the sample units from which components are missing (Kaski 1997).

All the experiments have been carried out on a PC computer with an Intel Pentium PIII-500 using MATLAB software with a program file written by the authors. Depending on the size of the input dataset, the training process can last from a few minutes to several hours, but this process has to be carried out only once. The U-matrix computation and the different displays last a very short time (a few seconds).

3.5 Conclusion

We presented in this paper some ways to use SOMs for visualizing an abundance dataset. Due to its extreme adaptability, the SOM can have a number of variants that make it a very convenient tool for studying the ecological communities.

The SOM enhanced by the U-matrix method is an effective clustering method including techniques to display the species abundance or abiotic variables.

The SOM is a promising approach and completes the results obtained by classical methods of classification.

References

Chon TS, Park YS, Moon KH, Cha EY (1996) Patternizing communities by using an artificial neural network. Ecological Modelling, 90, 69-78

Céréghino R, Giraudel JL, Compin A (2001) Spatial analysis of stream invertebrates distribution in the Adour-Garonne drainage basin (France), using Kohonen self organizing maps. Ecological Modelling, 146(1-3):-180

Foody GM (1999) Applications of the self-organizing feature map neural-network in community data-analysis. Ecological Modelling, 120(2-3) 97-107

Giraudel JL, Aurelle D, Lek S (2000) Application of the self-organizing mapping and fuzzy clustering microsatellite data: how to detect genetic structure in brown trout (*Salmo trutta*) populations. In Lek S and Gueguan J F (editors) Artificial Neuronal Networks: application to ecology and evolution, pages 187-201. Springer, Berlin, Heidelberg

Hämäläinen A (1994) A measure of disorder for the self-organizing map. In Proc. ICNN'94, Int. Conf. on Neural Networks, pages 659-664, Piscataway, NJ. IEEE Service Center

Kaski S, March E (1997) Data exploration using self-organizing maps. Acta Polytechnica Scandinavica, Mathematics, Computing and Management in Engineering Series No. 82

Kiviluoto K (1996) Topology preservation in self-organizing maps. In ICNN 96. The 1996 IEEE International Conference on Neural Networks, volume 1, pages 294-9. IEEE, New York, NY, USA

Kohonen T (1982) Analysis of a simple self-organizing process. Biol. Cyb., 44(2)135-140

Kohonen T (1995) Self-Organizing Maps, volume 30 of Springer Series in Information Sciences. Springer, Berlin, Heidelberg. 362 p

Kraaijveld MA, Mao J, Jain AK (1995) A nonlinear projection method based on Kohonen's topology preserving maps. IEEE Trans. on Neural Networks, 6(3)548-59

Lek S, Giraudel JL, Guégan JF (2000) Neuronal networks: Algorithms and architectures for ecologists and evolutionary ecologists. In Lek, S. and Gueguan, J.F., editors, Artificial Neuronal Networks: application to ecology and evolution, pages 3-27 Springer, Berlin, Heidelberg.

Lek S, Guegan JF (1999) Artificial neural networks as a tool in ecological modelling, an introduction. Ecological Modelling, 120(2-3)65-73

Ludwig JA, Reynolds JF (1988) Statistical Ecology, a primer on methods and computing. John Wiley & Sons

Merkl D, Rauber A (1997) Alternative ways for cluster visualization in self-organizing maps. In Proceedings of WSOM'97, Workshop on Self-Organizing Maps, Espoo, Finland, June 4-6, pages 106-111. Helsinki University of Technology, Neural Networks Research Centre, Espoo, Finland

Miikkulainen R (1990) Script recognition with hierarchical feature maps. Connection Science, 2, 83-101

Orlóci L (1978) Multivariate Analysis in Vegetation Research. W. Junk, The Hague

Peet RK, Loucks OL (1977) A gradient analysis of southern Wisconsin forests. Ecology, 58, 485-499

Ultsch A, Siemon HP (1990) Kohonen's self organizing feature maps for exploratory data analysis. In Proc. INNC'90, Int. Neural Network Conf., pages 305-308, Dordrecht, Netherlands. Kluwer

Van Tongeren OFR (1995) Cluster analysis. In Jongman RHG, Ter Braak CJF and Van Tongeren OFR (editors) Data analysis in community and landscape ecology, chapter 6, pages 174-212. Cambridge University Press

Vesanto J, Himberg J, Siponen M, Simula O (1998) Enhancing SOM based data visualization. In Proceedings of the International Conference on Soft Computing and Information/Intelligent Systems (IIZUKA'98), pages 64-67, Iizuka, Japan

Watson DF (1992) Conturing: a guide to the analysis and display of spatial data, volume 10 of Computer methods in the Geosciences. Pergamon Press

Whittaker RH (1952) A study of summer foliage insect communities in the Great Smoky Mountains. Ecological Monographs, 22, 1-44

Chapter 4

Ecological Applications of Genetic Algorithms

D. Morrall

4.1 Introduction

In the early 1960's biologists were attempting to simulate evolution in natural systems (e.g., Fraser 1960, 1962). About this same time, Holland was working towards the goal of expressing mathematically the adaptive processes of natural systems in order to create artificial systems using these processes. Holland's 1975 publication "Adaptation in natural and artificial systems" provides a landmark conceptual framework for evolutionary adaptation in artificial systems using genetic algorithms. His work on genetic algorithms was based on the premise that natural evolution offers the best model for balancing efficiency and flexibility in complex systems.

During the 1970's genetic algorithms were primarily the domain of computer programmers. Programmers studying artificial intelligence techniques were exploring design attributes of GAs such as mutation and crossover rates, model behavior, and overall performance. Goldberg (1989) provides a thorough overview of the development of genetic algorithms between the 1950's and 1980's. By the 1980's and 90's, genetic algorithms were becoming widely used as an optimization tool for a variety of real-world applications. Optimization techniques exploited the standard binary GAs cability to represent and optimize real-world problems. Some of the most common applications were in the area of combinatorial optimization. Combinatorial optimization models included the classical traveling salesman problem (see Goldberg 1989), worker scheduling (Carnahan et al. 2000), traffic flow (Srinivasan et al. 2000), electrical and waste routing (Savic and Walters 1997; Song et al. 1997; Rauch and Harremoes 1999; Su and Lii 1999) and molecular design (Hibbert 1993; Venkatasubramanian et al. 1995). Parametric optimization was somewhat less common than combinatorial optimization. Examples include growth media optimization (Weuster-Botz et al. 1995), drug release formulation (Hirsch and Muller-Goymann 1995), and optimization of bioprocess rates (Park et al. 1997).

Classifier systems and control strategies could perhaps be considered the next generation of GAs. While they are also optimization problems in some sense, they typically require a more complex structure than the bit-string GA. These applications often incorporate rules or symbolic capabilities. Pattern recognition (e.g., Lavine et al. 1999), equation discovery (D'Angelo et al. 1995), consumer

choice (Greene and Smith 1987), fighter plane combat (Smith et al. 2000), medical diagnosis (Pattichis and Schizas 1996) and various game playing programs are examples of classifier and control strategies. During the 1990's a shift towards hybrid approaches (e.g., GA-Neural Net combinations) began to emerge (Fishman and Barr 1991). Hybrid models capitalize on the GAs ability to evolve components of the hybrid system such as cellular automata or neural net node weightings.

Although simulations of evolution in natural systems were initiated in the field of biology, GAs were not widely used for ecological modelling until the 1990's. The development of the more complex GA designs that Goldberg (1989) refers to as genetic based machine learning paved the way for ecological modelling with genetic algorithms. This paper will explore the state of the art of GAs in the field of ecology and possibilities for future exploration of ecological systems using evolutionary programming. Though the focus of this paper is on genetic algorithms, many of the ideas presented in this paper are applicable to other machine learning and hybrid approaches that are based on the theory of evolution. The term evolutionary algorithm is sometimes used as a more general descriptor to refer to the entire suite of methods that employ the process of evolution through natural selection and will also be used in this text.

4.2 Ecology and Ecological Modelling

Eco- comes from the Greek "oikos" meaning home. "Ecology… is concerned with the most complex level of biological integration. It attempts to explain why organisms live where they do and what physical and biological variables govern their distribution, numbers, and interactions. Based on an understanding of the fundamental principles that govern organism distributions, numbers, and interactions, the future behavior and assemblages of organisms can be predicted. Induction is defined as arriving at knowledge of the universal from examination of particulars; to see what is common to a set of similars. Aristotle stated that "It is by induction that we know universals and the primary premises on which demonstrations are based" (Lloyd 1968). Ecology is largely a field of induction. The complexity of ecological systems and the vast array of interdisciplinary data that must be collected to understand an ecological system make the process of "arriving at knowledge of the universals from the particulars" a challenging endeavor.

To further complicate matters, non-linear relationships and dependencies in data are common in ecological systems, which are by definition integrative. Eugene Odum, the father of ecology, made the oft quoted statement that ecosystems are more than the sum of their parts; inferring the presence of non-linear interactions which can lead to unexpected emergent properties. Ecological modelling has, since its inception, attempted to develop ways to mathematically describe these complex, non-linear ecological systems. Traditional ecological

modelling can be thought of as a top-down or deductive technique that represents broad ecological principles to produce the detailed patterns observed in nature. Early classic models that have contributed greatly to ecological theory included the Lotka-Volterra (Lotka 1925, Volterra 1926) predator-prey population model and H.T. Odum's (1957) Silver Spring ecosystem energy flow model. These and other traditional ecological models are procedural, equation-based models. State variables, equations, and parameters are explicitly coded into the model. Equations and parameters are typically static with only the state variables changing over time. These types of models have done much to help us test ecological theory by determining if the observed system pattern could be produced by equations representing the biological processes and interactions. An ability to recreate the reflection tells us that the proposed interactions and mechanisms are plausible. These models paved the way for predictive models designed to simulate how the effect of a stressor (e.g., nutrient loading) might change the environment.

Major advances have been made in the field of ecology through ecological modelling. However, the discovery of new theoretical breakthroughs via traditional ecological modelling has been limited in the past decade. Jorgensen, in his paper on the state-of-the-art of ecological modelling (1999), suggests that ecological modelling has two primary difficulties that limit its effectiveness: obtaining reliable parameters and how to build ecosystem properties into the models. Constructing an ecosystem model requires a detailed understanding of ecosystem function in order to determine the appropriate level of complexity. Yet, even with reliable parameters and good model structure, traditional ecosystem models don't represent system properties of adaptation. As a result, ecosystem models base their analysis on parameters and structure at time t but attempt to predict ecosystem function at time t+1 (Jorgensen 1999). This failure to incorporate the dynamic structural aspects of ecological systems into our models limits both our ability to understand governing mechanisms and our ability to develop predictive models.

GAs are a bottom-up or inductive technique that can, through dynamic evolution, build the rules governing ecological systems. They offer a tool for parameter optimization and for the development of structurally dynamic models. Genetic algorithms can be used to solve a wide variety of problems but are most commonly employed when: you don't know what set of instructions to give to the computer to solve the problem (i.e., let the GA figure out the rules for you) the dataset is very large and an exhaustive search for the solution is inefficient (e.g., traditional optimization techniques won't work) the data are fuzzy or there is missing information the response surface is irregular (i.e., you need to find a general solution) Koza (1992) goes as far as to say that genetic programming provides "… a single, unified, domain dependent approach to the problem of induction" (Figure 4.1).

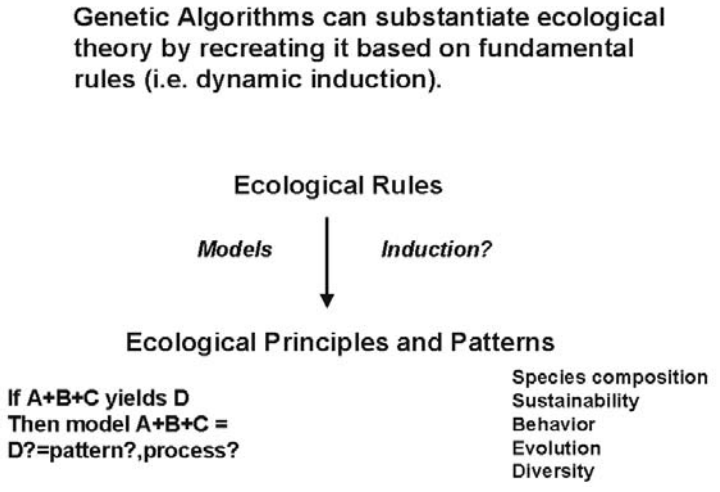


Figure 4.1. Ecological concepts are discovered from complex data by the process of induction. Genetic algorithms provide a mathematical tool to guide the researcher through the inductive process.

4.3 Genetic Algorithm Design Details

Genetic Algorithms (GA) are computer solution-search and problem-solving techniques based on the principles of evolution by natural selection. Through the process of natural selection, GAs evolve linear, coded representations of data to solve problems or develop strategies. Selection rules are designed by the programmer to govern the direction that is taken to evolve solutions to problems. Control strategies may be defined based on the programmer's conceptions about how a system operates (e.g., select organisms that are better at procuring food) or rules may be independent of internal system operation (e.g., optimize for correlation between observed and predicted organism distributions). Rules may be altered simply to produce a desired outcome (i.e., without any implied causality) or to evaluate multiple hypotheses about how a system operates.

Although the concept of GA is simple, actual GA development involves multiple design decisions and choices from among a wide variety of implementation techniques (see Holland 1975 for a classic example). The optimization problem is coded as a finite-length string, often using a binary representation. Therefore, it must be possible to represent the solution as this finite-length string. Problem representation by a fixed-length string is the key limitation to GAs. For more detail on GAs see Goldberg (1989). The basic strategy includes the following procedures (Figure 4.2). GAs randomly create a population of individuals. The mathematical representation of each individual

depends upon problem to be solved. Individuals in the population are evaluated to determine their fitness (e.g., how well the individual produces the desired outcome). The fitness evaluation is the most important determinant of how future recombination is guided and the direction in which the population will evolve. After fitness has been determined, individuals must be chosen to be parents. There are a variety of methods that can be used for parent selection. Tournament selection and roulette are 2 common methods. Once the parents have been chosen, each parent is cloned to produce a child that is an exact replica. Genetic material of the children is then exchanged via crossover or altered via mutation.

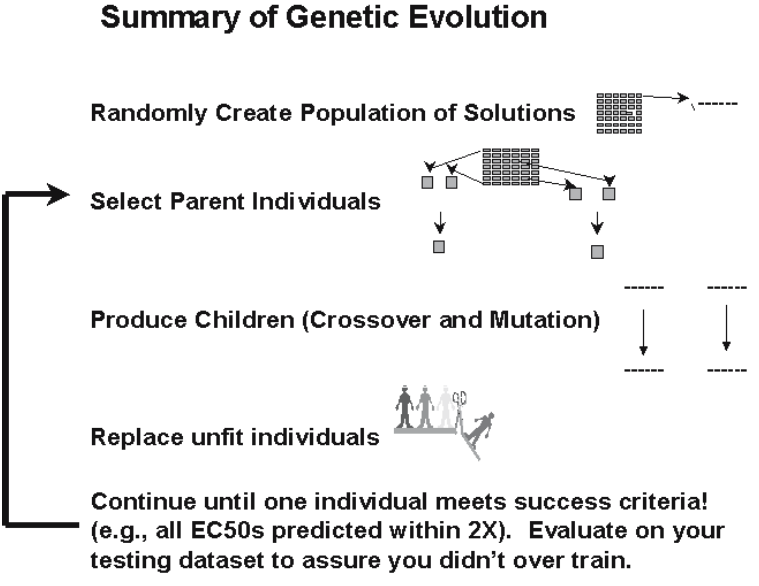

Summary of Genetic Evolution
Randomly Create Population of Solutions
Select Parent Individuals
Produce Children (Crossover and Mutation)
Replace unfit individuals
Continue until one individual meets success criteria! (e.g., all EC50s predicted within 2X). Evaluate on your testing dataset to assure you didn't over train.

Figure 4.2. Summary of the process of evolution with a genetic algorithm. The population of size n is randomly created where each individual is a chromosome constructed of genes (e.g., genetic material). Parents are selected from the population and reproduce to create children. The children undergo mutation and crossover. Their fitness is evaluated and the unfit individuals in the population are replaced. The process continues until a desirable outcome is achieved.

The probabilities of mutation and crossover are selected by the user. The children replace the least fit individuals in the population. The process continues either until a specified number of generations have been completed or until an individual in the population meets the success criteria. Because there is a random component in the GA, no two runs are the same. The trajectory of population fitness over time will vary between runs (Figure 4.3).

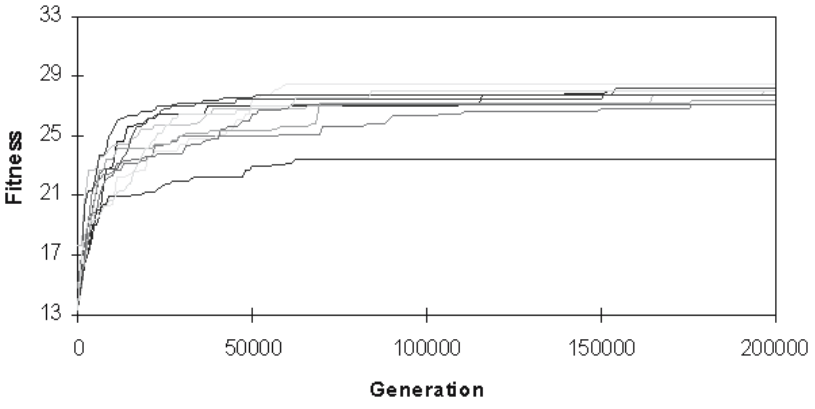


Figure 4.3. Example of the variability in fitness scenarios between different genetic algorithm simulations using the same parameter set. The genetic algorithms was run 10 times using the same set of parameters. The difference in fitness scenarios is a result of the random initialization of the population and the random nature of the mutation and crossover operators.

4.4 Applications of Genetic Algorithms to Ecological Modelling

In the field of ecology as in other fields, genetic algorithms have been used for parameter optimization, equation discovery, and pattern searching. Reynolds and Ford (1999) developed a modelling approach (Pareto Optimal Model Assessment Cycle) that allows for simultaneous evaluation of multiple output criteria for a model with a given parameter set. A genetic algorithm is used to generate optimal parameter sets (Pareto Optimal Sets) and evaluate fitness in terms of their ability to predict the model output. Reynolds used this approach to parameterize and evaluate the ecological theory, model structure, and assessment data of WHORL, a canopy competition model. Through this dynamic process they were able to reveal deficiencies in the model structure and criteria to make improvements. Ludvigan et al (1997) used a GA to search for optimal bacterial phospholipid fatty acid (PLFA) combinations to biogeochemical parameters. Random combinations of PLFAs were chosen and evolved using a GA with a partial least squares fitness function. These combinations of engineering and statistical techniques with evolutionary algorithms provide a robust approach to data evaluation.

Similar in approach to parameter optimization is the use of GAs to look for data patterns such as subsurface zones of bacterial activity (Mahinthakumar al. 1999) and fish distributions (D'Angelo et al. 1995). These studies attempt to find the optimum combination of variables to predict the distributions of organisms in

relation to their habitat. With the surge in use of geographic information systems (GIS) to present environmental data on a spatial template, these types of analysis are common and are expected to become more prevalent. Jeffers (1999) and Stockwell (1999) provide examples of some species distribution models that use abundance or presence/absence data to predict distributions of organisms as a function of spatial habitat characteristics. Fielding (1999) suggests that these types of exploratory analysis are some of the most promising and least controversial uses of machine learning applications.

From a practical standpoint, GAs have several limitations in the way that they have been used historically in ecology. Many of the current GA applications can be accomplished using other more traditional techniques. Often the traditional techniques don't perform as well as a GA (e.g., applications of linear statistics to non-linear problems), however, they always beg the question "why didn't you use technique X ?". Also, GAs are not very transparent to non-users. While it is easy to explain the basics of how a GA operates, it still seems like magic to the uninitiated. And finally, many of the combinatorial, parametric and pattern recognition applications of GAs require large datasets.

With these limitations in mind, it seems probable that new ground will be broken by GAs that do things for which GAs are uniquely suited. For example, GAs that incorporate evolution into ecological systems can allow us to truly explore the science of evolution, to understand ecosystems and open the doors to a multitude of practical applications. These GAs are distinct from earlier process models in that they can evolve both the equations governing them and the equation rate parameters. Through the GA, the modeler produces a dynamic "movie" of the evolution of mechanisms producing the patterns. Questions can be posed regarding whether and how a particular mechanism might have arisen to produce a natural phenomenon. Typically few assumptions and constraints, other than Darwinian selection, are built into a genetic algorithm model. As a result, GAs can substantiate ecological theory by recreating it based on fundamental rules (i.e., dynamic induction).

There are some fascinating examples of explorations into evolutionary theory and the emergence of ecological systems and properties. The fundamentals of how to build these models came largely from computer game playing strategies (e.g., Bouskila et al. 1998). Koza (1992) in his book on genetic programming presents a thorough discussion of a genetic program designed to find the optimal foraging strategies for an *Anolis* lizard and emergent properties in ants. In the lizard example, the optimal foraging strategy as described by Roughgarden (1992) was discovered based simply on the location of the prey, the abundance of the prey, and the velocity of the lizard. Over time, the lizard can adapt his strategy if the environmental conditions change. This provides both an example of control-cost strategy and a demonstration that evolutionary programs can evolve the theories governing complex behavioral strategies using only simple governing rules. In the ant emergent behavior program (Deneubourg et al. 1986; 1991), a set of rules is used that governs the actions of individual ants. When these rules are simultaneously executed, a complex pattern of behavior emerges that causes the ants to collect and consolidate food pellets into a single pile.

Researchers have long hypothesized that collective behavior could arise from colonial organisms operating according to very simple rules. Traditional engineering style models would require tremendous computer power to simulate individuals and explicitly code the possible sets of interactions. Genetic algorithms, on the other hand, are especially well suited for this type of applications and demonstrated that the emergence of complex behavior was indeed possible from simple rules. Others, including Kvasnicka and Pospichal (1999) and Reuter and Breckling (1999) have similarly demonstrated emergent behavior among artificial agents and organisms. GAs, such as Giske et al.'s (1998) model of spatial dynamics in fish, that simulate coordinated movement of organisms based on simple rules have greatly improved our understanding of how groups of organisms travel, sometimes great distances, together in a coordinated fashion.

The transition from population and community ecology to ecosystem science occurs when organisms are placed in a physical and chemical environment. Ecosystem science deals with the interaction between organisms and their non-living environment. An important component of the ecosystem is the spatial landscape within which organisms live. Natural landscapes have complex patch dynamics that influence both the distribution of organisms and their interactions (Borhman and Likens 1979). The evolution of multiple-species within a landscape requires representation of diverse niches with different evolutionary pressures (Cedeno and Vemuri 1999). Organisms may move through the system and change in number as well as immigrate and emmigrate as system properties change. It is through their ability to simulate evolution in response to changing environmental conditions and spatial heterogeneity that GAs can offer most to the field of ecosystems ecology.

One example of a genetic algorithm that can be used to simulate evolution in ecosystems is Holland's Echo model (1995). Echo is a generic ecosystem model with evolving agents and a resource limited environment. Hraber et al. (1997) state that the primary contribution of Echo to ecological modelling is that evolution is built in as a fundamental part of the system. Echo includes both ecological interactions and evolutionary dynamics. This provides the potential for evolution of ecological structure and function in response to changing conditions. Through this approach they can simultaneously evolve multiple species and species interactions. They consider Echo a mechanistic model because primitive components and mechanisms are built into the model that spontaneously give rise to macro-level properties. Echo incorporates spatial attributes of the ecosystem and the opportunity for co-evolution; both of which are essential for simulation of complex ecosystems. Agents, or individuals within Echo, occupy sites within a two-dimensional world. They reproduce and exchange genes when they have acquired sufficient resources through trade and combat. Each agent contains 6 external tags (offense, defense, and mating) and internal conditions (combat, trade, and mating) genes (Figure 4.4). Internal tags are not visible to other agents. Agents interact based on their own internal conditions and the other agents' external tags. They have different abilities to accept and accumulate resources. Like Echo, EUZONE (Downing 1997) describes an evolutionary computation

model that includes both ecological and evolutionary interactions. EUZONE evolves species of phytoplankton-like creatures in a two-dimensional world. Echo and EUZONE are designed to capture the fundamental attributes of complex adaptive systems.

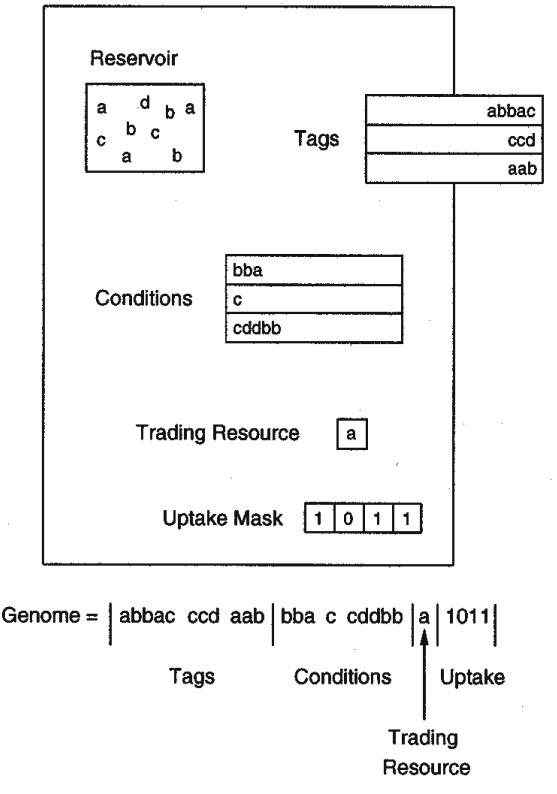


Figure 4.4. The structure of an Echo agent. Each agent consists of a gnome and a resource reservoir. The genome has r+7 genes, where r is the number of resources in the world. Six of these, the tags and the conditions, are composed of variable-length strings or resources (i.e., of the lower case letters that represent resources). Tags are visible to other agents. Conditions and other properties are not.

GAs can also be used to develop self-designing ecosystems. The development of self-designing adaptive systems solves the problem that process models have in trying to represent dynamically changing systems. Fontaine (1981) made an early attempt at creating a self-designing ecosystem using a standard process model and adjusting the parameters. The model parameters were optimized by running the model for 3 time-steps, altering the parameters and rerunning the model with the

best set of parameters. The goal of this effort was "the search for a single principle that adequately describes the evolution of ecosystem structure and function at all levels of resolution…" (Fontaine 1981). With evolutionary programming not only can the parameters be optimized more effectively, but the model structure, state values, and behaviors can be changed to produce a dynamic, evolving ecosystem. GAs and GPs were developed to create self-designing computer programs (Koza 1992) and are therefore the logical techniques to be applied to this effort. Self-designing computer programs can be used to create self-designing ecosystems. The work of Holland (1995) and Hraber et al. (1997) are in essence self-designing systems.

4.5 Predicting the Future with Genetic Algorithms

Sustainability has become the ultimate goal of many ecological management practices in the 21st century. Regardless of what we are attempting to sustain it must be sustained in the face of a continuously changing environment. Natural change is accelerated through human alteration of habitats (e.g., channelization of streams), addition of pesticides and fertilizers, and the introduction of non-native species. For organisms to survive they must be able to adapt to both natural and human-induced change. Ecological models must also incorporate evolution and co-evolution into their frameworks if they are to predict the sustainability of various ecologies (e.g., Janssen 1998). Evolution can be incorporated into ecological models through adaptation of species currently in the system and by forecasting changes in species composition (e.g. Maier et al. 1998). Co-evolution in ecological systems is described by the Gaia theory (Lovelock and Margulis 1974; Downing and Zvirnsky 1999) and which incorporates the feedback mechanisms common in natural systems. Gaia refers to the circular pathway whereby organisms respond to their environment, modify the environment, and are, in-turn, modified by the environment. Representation of this phenomenon is critical for predicting the future of ecological systems.

As Jorgensen (1999) noted, most traditional models are limited because they use a static representation of ecological systems that is developed and parameterized based on the system characteristics at a certain point in time. They do not take into account the flexibility and adaptive capabilities of natural systems and therefore may over-predict or erroneously predict the effect of stressors. A combination of genetic algorithms with traditional engineering based models and other artificial intelligence techniques (e.g., cellular automata and neural networks) can provide a dynamic representation of how adaptive responses to environmental change govern species change. These dynamic approaches facilitate exploration of various possible trajectories of adaptation that might result from changes in the environment.

Because of the broad scale at which the environment is being changed, tools are needed that can accurately predict the sustainability of populations, communities,

and ecosystems. This need will become increasingly important in the future. Dynamic simulation of organism adaptation and interdependencies marks the beginning of a new age in ecological modelling and offers the possibility for development of superior predictive models. GAs based on the fundamentals of theoretical ecology can help us find the mathematical foundations for the concepts on which predictive ecology is based.

4.6
The Next Generation: Hybrid Genetic Algorithms

The examples noted above point out the capabilities of GAs to optimize parameters, discover equations, search for patterns, and develop classifier and control strategies. Yet GAs are not the best technique for all problems. Neural nets, for example, are often superior at pattern recognition. Many readily available statistical techniques perform as well as or better than GAs at developing regression and classification systems. These techniques, however, do not have the evolutionary capabilities needed to develop control strategies and complex adaptive systems.

While each of these techniques will continue to be used very successfully, hybrids using the best attributes of each technique will have the potential to offer revolutionary advances. Work done by Patel et al. (1998) is an intriguing example of a hybrid neural net and genetic algorithm. Their objective was to design new molecules that would kill bacteria. They trained a neural network on a dataset of chemicals and their ability to kill bacteria. They then used a genetic algorithm to rearrange the amino acid sequences to create new chemicals. The neural network then evaluated the efficacy of these new chemicals. This approach took advantage of the strengths of each technique.

D'Angelo-Morrall et al. (in prep) created a hybrid statistical clustering and GP model. The objective of this work was to predict the toxicity to aquatic organisms of a wide variety of chemicals. Because the chemicals in the dataset were very diverse, a single equation could not adequately predict the toxicity of all chemicals. They used k-means clustering to group the chemicals into similar classes. The GP was then used to evolve equations to predict the toxicity of each groups of chemicals. The performance of the hybrid statistical/GP was compared with a GP that did both clustering and predictions and K-means analysis that did both clustering and predictions. When the dataset was of sufficient size for predictions (training set n=50; testing set n=10) the hybrid approach outperforms the individual methods.

Whigham and Recknagel (2001) developed a novel hybrid of a process-model and GA with the goal of optimizing the model structure. They started with a standard process model and used the GA to optimize the process-based equations and, where necessary, evolve new equations. This hybrid model performed better than either a stand-alone GA or process model. This type of hybrid has great potential for producing better process models and provides a means by which we

can question the many of the standard process equations that have become paradigms.

These are just a few examples of hybrid architectures, which are becoming more and more prevalent. It is expected that in the future, the powerful evolutionary framework of GAs will be commonly used as an integral component of hybrids. This transition from independent frameworks to coupled systems is the next generation in the evolution of artificial intelligence programming techniques and has the potential to open new frontiers in ecological modelling.

References

Borman FH, Likens GE (1979) Pattern and process in a forested ecosystem. Springer-Verlag, New York

Bouskila A, Robinson ME, Roitberg BD, Tenhumberg B (1998) Life-history decisions under predation risk: importance of game perspective. Evolutionary Ecology, 12: 701-715

Carnahan BJ, Redfern MS, Norman B (2000) Designing safe job rotation schedules using optimization and heuristic search. Ergonomics, 43: 543-560

Cedeno W, Vemuri VR (1999) Analysis of speciation and niching in the multi-niche crowding GA. Theoretical Computer Science, 229: 177-197

D'Angelo DJ, Howard LM, Meyer JL, Gregory SV, Ashkenas LR (1995) Ecological uses for genetic algorithms: predicting fish distributions in complex physical habitats. Can. J. Fish. Aquat. Sci., 52: 1893-1908

Deneubourg JL, Aron S, Goss S, Pasteels JM, Deurinck G (1986) Random behavior, amplification processes and number of participants: How they contribute to the foraging properties of ants. In D. Farmer, A. Lapedes, N. Packard, and B. Wendroff (editors), Evolution, games, and Learning. North-Holland

Deneubourg JL, Goss S, Franks N, Sendova-Franks A, Detrain C, Chretien L (1991) The dynamics of collective sorting robot-like ants and ant-like robots. In: J. Meyer and S. Wilson (editors), From Animals to Aniats: Proceedings of the First International Conference on Simulation of Adaptive Behavior. MIT Press

Downing K (1998) Using evolutionary computational techniques in environmental Modelling. Environmental Modelling and Software, 13: 519-528

Downing K, Zvirinsky P (1999) The simulated evolution of biochemical guilds: reconciling Gaia theory and natural selection. Artificial Life, 5: 291-318

Fielding AH (1999) Machine learning methods for ecological applications. Kluwer Academic Publishers, Boston, Massachusetts, 261 pp

Fishman MB, Barr DS (1991) A hybrid system for market timing. Artificial Intelligence, August, 26-34

Fontaine TD (1981) A self-designing model for testing hypotheses of ecosystem development 281-291. In: D. Dubois (editor), Progress in ecological engineering and management by mathematical Modelling, Proc. 2nd Int. Conf., State-of-the-Art of Ecological Modelling, 18-24 April 1980, Liege, Belgium. P. 720

Fraser AS (1960) Simulation of genetic systems by automatic digital computers. 5-linkage, dominance, and epistatis. In: O. Kempthorne (editor), Biometrical genetics, Macmillan, New York, pp. 70-83

Fraser AS (1962) Simulation of genetic systems. Journal of Theoretical Biology, 2: 329-346

Giske J, Huse G, Fiksen O (1998) Modelling spatial dynamics in fish. Reviews in Fish Biology and Fisheries, 8: 57-91

Goldberg DE (1989) Genetic algorithms in search, optimization and machine learning. Addison-Wesley, Reading, Massachusetts, 412 pp

Green DP, Smith SF (1987) A genetic system for learning models of consumer choice. In: J.J. Grefenstette (editor), Genetic algorithms and their applications: Proceedings of the second annual conference on Genetic algorithms. Lawrence Erlbaum Associates Publishers, Hillsdale, New Jersey

Hibbert DB (1993) Generation and display of chemical structures by genetic algorithms. Chemometrics and Intelligent Laboratory Systems, 20: 35-43

Hirsch R, Mueller-Goymann CC (1995) Fitting of diffusion coefficients in a three-compartment sustained release drug formulation using a genetic algorithm. International journal of pharmaceutics, 120: 229-234

Holland JH (1975) Adaptation in natural and artificial systems. Ann Arbor, The University of Michigan Press

Holland JH (1995) Hidden Order: How adaptation builds complexity. Addison-Wesley, Reading, 185 p

Hraber PT, Jones T, Forrest S (1997) The Ecology of Echo. Artificial Life, 3: 165-190

Janssen M. (1998) Use of complex adaptive systems for modeling global change. Ecosystems, 1: 457-463

Jeffers JNR (1999) Genetic Algorithms I. In A. H. Fielding (editor) Machine Learning Methods for Ecological Applications. Kluwer Academic Press, Massachusetts, 261pp

Jorgensen SE (1999) State-of-the-art of ecological Modelling with emphasis on development of structural dynamic models. Ecological Modelling, 120: 75-96

Koza JR (1992) Genetic Programming. Massachusetts Institute of Technology, Cambridge, Massachusetts 810 pp

Kvasnicka V, Pospichal J (1999) An emergence of coordinated communication in populations of agents. Artificial Life, 5: 319-342

Lavine BK, Moores AJ, Mayfield H, Farugue A (1999) Genetic algorithms applied to pattern recognition analysis of high-speed gas chromatograms of aviation turbine fuels using an integrated Jet-A/JP-8 database. Microchemical Journal, 61: 69-78

Lloyd GER (1968) Aristotle, The growth and structure of his thoughts. Cambridge University Press, Cambridge, Massachusetts

Lotka AJ (1925) Elements of Physical Biology, Williams & Williams, Baltimore

Volterra V (1926) Variazione e fluttazioni de numero d'individiu in specie animali conviventi. Translated in R.N. Chapman, 1931, Animal Ecology, McGraw-Hill, New York

Lovelock J, Margulis L (1974) Atmospheric homeostatis by and for the biosphere: The Gaia hypothesis. Tellus, 26: 2-10

Ludvigsen L, Albrechtsen JJ, Holst H, Christsen TH (1997) Correlating phospholipid fatty acids (PLFA) in a landfill leachate polluted aquifer with biogeochemical facors by multivariate statistical methods. FEMS Microbiology Review, 20: 447-460

Mahinthakumar G, Gwo JP, Moline GR, Webb OF (1999) Subsurface biological activity zone detection using genetic search algorithms. J. Env. Engr. –ASCE, 125: 1103-1112

Maier HR, Dandy GC, Burch MD (1998) Use of artificial neural networks for modelling cyanobacteria *Anabaena* spp. in the River Murray, South Austrailia. Ecol. Modelling, 105: 257-272

Morrall DD, Billenstein M, Miagkikh V, KariKari A (in prep.) A hybrid k-means GP for predicting the toxicity of chemicals to aquatic organisms

Odum HT (1957) Trophic structure and productivity of Silver Springs, Florida, Ecol. Monogr., 27: 55-112

Park LJ, Park CH, Park C, Lee T (1997) Application of genetic algorithsm to parameter estimation of bioprocesses. Medical and biological engineering and computing, 35: 47-49

Patel S, Scott IP, Bhakoo M, Elliott P (1998) Patenting computer-designed peptides. Journal of Computer-Aided Molecular Design, 12:543-556

Pattichis CS, Schizas CN (1996) Genetics-based machine learning for the assessment of certain neuromuscular disorders. IEEE Transactions on Neural Networks, 7: 427-439

Rauch W, Harremoes P (1999) Genetic algorithms in real time control applied to minimize transient pollution from urban wastewater systems. Wat. Res., 33, 1265-1277

Reuter H, Breckling B (1999) Emerging properties on the individual level: Modelling the reproduction phase of the European robin. Ecol. Modelling, 121:199-219

Reynolds JH, Ford ED (1999) Multi-criteria assessment of ecological process models. Ecology, 80: 538-553

Roughgarden J (1992) *Anolis* lizards of the Caribbean: ecology, evolution, and plate tectonics. Oxford University Press

Savic DA, Walters GA (1997) Genetic algorithms for least-cost design of water distribution networks. Journal of water resources planning and management, March/April 67-77

Smith RE, Dike BA, Mehra RK, Ravishandran B, El-Fallah A (2000) Classifier systems in combat: two-sided learning of maneuvers for advanced fighter aircraft. Computer methods in applied mathematics and engineering, 186: 421-437

Song YH, Wang GS, Wang PY, Johns AT (1997) Environmental/economic dispatch using fuzzy logic control genetic algorithms. EII Proc.-Gener. Transm. Distrib., 144: 377-392

Srinivasan D, Cheu RL, Poh YP, Ng AKC (2000) Development of an intelligent technique for traffic network incident detection. Engineering applications of artificial intelligence, 13: 311- 322

Stockwell DRB (1999) Genetic Algorithsm I. In A. H. Fielding (editor) Machine Learning Methods for Ecological Applications. Kluwer Academic Press, Massachusetts, 261pp

Su CT, Lii GR (1999) Reliability planning employing genetic algorithms for an electrical power system. Applied artificial intelligence, 13: 763-776

Tomassini M (in press) A survey of genetic algorithms. Annual Reviews of Computational Physics, World Scientific, Volume III

Venkatasubramanian V, Chan K, Caruthers JM (1995) Evolutionary design of molecules with desired properties using the genetic algorithm. J. Chem. Inf. Comput. Sci,. 35: 188-195

Weuster-Botz D, Pramatarova V, Spassov G, Wandrey C (1995) Use of a genetic algorithm in the development of a synthetic growth medium for Arthrobacter simplex with high hydrocortisone Δ^1 – dehydrogenase activity. J. Chem. Tech. Biotechnol, 64: 386-392

Whigham P, Recknagel FA (2001) Predicting chlorophyll-a in freshwater lakes by hybridising process-based models and genetic algorithms. Ecological Modelling 246, 1-3, 243-252

Chapter 5

Ecological Applications of Evolutionary Computation

P.A. Whigham · G.B. Fogel

5.1 Introduction

Ecological modelling covers a broad range of techniques, concepts and fields of study. Ecosystems display many complex structures, such as hierarchical organization, and many components interacting at different temporal and spatial scales. These components, and the overall structure of any ecosystem, have also been derived through complex adaptations that produce such features as speciation, symbiosis and community structures.

The most basic ecological models are concerned with the behavior of a single species, group of species or community and their variation in time. These models can be expressed as predictive functions, based on independent variables such as environmental factors, other species and other abiotic factors. More complex models capture the coupled interaction between species and may incorporate spatial extent when describing the system. The most complex models attempt to capture the spatial and temporal patterns of an ecosystem at several levels of description. These models aim to understand the biological and evolutionary mechanisms that have produced an observed ecosystem and to explore the basic properties that produce diversity and structure in an idealized system.

Evolutionary computation is a discipline that makes use of principals from natural evolution to evolve solutions to complex computational problems. These techniques have been applied to such diverse areas as optimization, inductive modelling, constructing characteristic features of biological systems and as theoretical models of social and population-based interactions. Their capability to handle large search spaces (in terms of possible decisions or model constructs) in an efficient manner, and their similarity to biological systems, such as genotype/phenotype mapping, structured populations of individuals, adaptation and evolution makes them ideal candidates for constructing ecological models.

The advent of technologies such as global positioning systems, satellite images, improved field data collection services and unobtrusive tracking devices have

allowed ecologists to collect data at new resolutions and with higher accuracy and frequency. This improved data collection offers the possibility for the use of inductive modelling techniques to develop new theories and models of ecological systems at many scales and complexities. This data can also be used to verify theoretical models of ecology and to give insight to the complex hierarchical structures inherent in ecosystems.

This chapter outlines some of the basic approaches to ecological modelling and shows that evolutionary computation techniques have been successful in a number of different ecological areas. The diverse range of techniques supported by evolutionary algorithms is shown to be appropriate for the extension of current approaches and the development of new techniques for understanding ecological systems, and is a major area of the emerging discipline of ecological informatics.

5.2 Ecological Modelling

Ecology is a discipline that covers many scales of description, both spatial and temporal, and is fundamentally concerned with the interactions between organisms and their environment (Gillman and Hails 1997). Haeckel (1866) first defined the concept of ecology in terms of an understanding of components (biotic and abiotic), and how they can be viewed, described and modelled as a system. An ecological model must be able to describe the changes in a system based on generalities of how a system is functioning, the selected components that make up the system, and within certain temporal and spatial resolutions. Ideally these models allow either a prediction in terms of future states of the system, or elicit greater understanding of how the system functions and the driving forces and interactions of the system. Ecology may also be defined as the study of the processes that influence the distribution and abundance of organisms, their interactions and the transformations of energy within a system.

5.2.1 The Challenges of Ecological Modelling

Ecological modelling is a challenging discipline due to the inherent complexity, nonlinearity, and spatiotemporal dynamics of the multivariate system being described. For example, computational models of individual interactions in ecological systems can be very complex (Judson 1994). These models are typically characterized by many parameters that describe how each individual behaves. As a result, the system as a whole produces complex, often unpredictable behavior. Alternative models of individuals produce a mathematical equation that relates several biotic and abiotic independent quantities to a particular dependent variable. This equation can then be used to predict the future states of the system

given present knowledge. These models are often formulated as a combination of deterministic factors combined with a term or terms that capture the intrinsic stochastic nature of the system and the errors associated with measurement and observation. Of course, although these models are typically expressed in linear terms, they describe a system that is generally replete with nonlinear interactions and temporal variation.

Extensions to these basic linear functions are often formulated as differential equations (when time is continuous) or difference equations (when time is taken in discrete intervals). Difference equations rely on an interpretation of nonlinear systems and their resolution in the form of a singular equation, and have been applied to individuals and population dynamics (Hassel and Comins 1976). With this approach, it may not be possible to encapsulate all of the correct variables into the equation or might incorrectly set the importance (weights) for each term in the equation. There are also a number of simplifying assumptions required when formulating difference or differential equations that can make the resulting models difficult to interpret when applying them to real ecosystems. Coupled equations have been used to understand and explore plant-herbivore, host-parasite, host-pathogen and other competition interactions. Extensions to these models have allowed intraspecies competition to be represented (for example, (Watkinson 1987)), however all these equations have constants that must be set to represent the corresponding modelled system. As these equations become more complex the issue of parameter selection, based on measured data, has become an issue due to the nonlinear behaviour of the model as a whole.

The dynamics of communities are interested in the interactions between species, and how the communities as a whole respond to changes in exogenous factors, and the introduction or removal of species in the community pool. These models are often constructed as a matrix of interactions, and expressed as partial differential equations. The dynamics of communities are often very complex and are difficult to model in any complete sense. Once again, the tuning of parameters and the selection of other constants in the model relies on matching the measured dynamics of the community and the model.

The previously described models have focused on the time dynamics of systems, and have assumed that the spatial interactions can be ignored, are at the same scale, or are homogeneous. However, since most ecological systems have spatial extent, and are limited in their interactions by location, the inclusion of space is often fundamental for exploring ecological systems. Extending the previous non-spatial models to a regular grid often involves extending the complexity of the model to include not only local interactions (at a single grid cell) but the influence of neighbouring cells and the species found at these locations. These models, often termed cellular automata (CA), have been widely used in studying plant and animal interactions (Comins et al. 1992; Silverton et al. 1992; Colasanti and Grime 1993). Spatial models often display long-term population persistence and dynamics that cannot be captured using a spatial models. Metapopulation models, based on spatial distributions, have also been

studied, where the local models are described using a differential or difference equation that incorporate a colonization term that indicates how local populations spread (Levins and Culver 1971; Carter and Prince 1981). These models differ from CA in that the spatial terms are explicitly represented in the equations, rather then an explicit model of spatial distribution being used to give spatial extension. The difficulty with cellular models is that the production of realistic local rules that give rise to appropriate global behaviour is a complex task with few guidelines. Spatial structure has also been studied in relation to island biogeography and the concept of gene flow between metapopulations (McCauley 1995), however all of these approaches are complex in nature and difficult to formulate and test.

5.2.2 Summary

The previous discussion presented some of the basic challenges involved in modelling ecological concepts. There are several salient points: models based on differential and difference equations have constants that require tuning based on measured data, and most models are simplified in order to be solved, either by assumptions that produce linear models, or by removing the complexities of the system. Additionally, models that incorporate spatial and temporal information, and those based on populations of individuals, are difficult to formulate or express. In the following sections, applications of evolutionary algorithms to ecological modelling will be reviewed. These applications are both extensions to traditional modelling efforts and wholly new approaches to ecological informatics.

5.3 Evolutionary Computation

Evolutionary computation (EC) is an area of computer and information science that uses principals from biological evolution to solve computational problems. The concept of simulating evolution on a computer has a long history. Early efforts focused on learning machines (Friedberg 1958; Friedberg 1959; Fogel 1962; Fogel et al. 1966), evolutionary systems dynamics (Barricelli 1954; Conrad and Pattee 1970), engineering applications (Rechenberg 1965; Schwefel 1965), and genetics (Fraser 1957; Bremermann 1962; Fraser 1962; Bremermann et al. 1966; Bagley 1967; Rosenberg 1967; Holland 1969; Holland 1973). This history is reviewed in Fogel (Fogel 1998; Fogel 2000). Since natural evolution has successfully created complex systems, and discovered novel solutions to difficult problems (such as vision, language and cooperation), there was a clear attraction towards the use of these principles to construct information systems that could be

used for modelling, discovery of patterns, construction of artificial systems, design work and optimization.

Evolution in real-world systems can best be described as a two-step process of heritable variation and selection. Variation occurs in the variety of behaviors that are exhibited by individuals of organisms interacting in communities, populations, and environments. The individual behavior (phenotype) is the product of a genetic composition (genotype) and the interaction of that genotype with the cellular environment. However, nature only measures the worth (fitness) of any individual at the level of phenotype. Selection removes those individuals from the population that do not have an appropriate fitness leaving behind those organisms with sufficient fitness to pass their genotype to the subsequent generation. During this process of heredity, variation to the genotype can occur, which may or may not lead to alternative behavior in the progeny. The process of selection repeats itself on the second generation of individuals, culling those with insufficient fitness. Variation in the reproductive process is the source of change at the genetic level, which may translate into new innovation at the phenotypic level. Selection serves as a filtering mechanism to ensure that individuals of low fitness are removed along the way. Evolution, then, is the coupling of these two processes over time.

Sewall Wright (1932) offered the concept of an adaptive landscape as a means to describe the manner in which evolution may proceed into novel adaptive zones. Individual genotypes can be mapped into their respective phenotypes, which are in turn mapped onto the surface of an adaptive topography. Each peak on this topography represents a phenotype of high fitness (and, therefore, one or more optimized genotypes). Evolution proceeds up the slopes of these peaks towards solutions of increasing fitness as the selective mechanism culls inappropriate phenotypic variants. However, this is an admittedly idealized concept. In reality, the adaptive topography changes with time as a function of the environment and organism-environment interactions. Simulation of evolution in a computer can demonstrate these same phenomena and can be used to search both static and temporal fitness landscapes for regions of high fitness.

Within an engineering context, search algorithms define a problem in terms of a search-space (the space of all possible solutions). Individual points in this search-space represent solutions to the problem at hand. The goal is to find useful solutions by traversing this search space in an efficient manner. However in many engineering problems, the number of potential solutions is astronomical and an exhaustive search of all solutions is infeasible in real time. Evolutionary algorithms have proven to be successful at searching complex nonlinear adaptive topographies (fitness functions) to return a near-optimal (or optimal) solution in real time. Evolutionary algorithms are an extremely successful approach to problems that can be framed as a search for a set of particular values, conditions or structures. A requirement for such a system is the ability to measure a relative fitness between individuals. Through the use of evolutionary algorithms, complex, nonlinear problems can be searched for near-optimal solutions. However, since evolution represents a stochastic search through the solution space, no application

can guarantee the optimal solution. However, for many problems a real-time near-optimal solution is quite satisfactory, and in fact often it is not possible to determine when an optimal solution has been reached. Evolutionary algorithms find good solutions to novel problems in complex situations – requirements that suit the concepts of ecology and ecological modelling.

5.3.1 The Basic Evolutionary Algorithm

The basic evolutionary algorithm (EA) comprises the following main components, independent of the actual representation used for individuals:

A method for generating an initial population of individuals. Typically this is made at random. The representation of the individual in the population is typically correlated to the problem that is being addressed.

A fitness function. This function gives a measure of fitness that can be used to score the worth of individuals in the population in terms of their performance to the task at hand.

A method of selection, based on fitness. This selection pressure drives the population towards better solutions. Common forms of selection are proportional, where the probability of selection is directly proportional to the fitness of an individual compared with the population as a whole, and tournament (round-robin), where the selection is based on a fitness ranking between a random subset of the population.

A method of reproduction with heritable variation. Reproduction may mimic various genetic operators, such as mutation and crossover, to produce new individual behavior in the population with some variety. These operators commonly allow for both random changes within an individual's representation, and the sharing of genetic information between two or more members of the population. Alternatively, the variation operators can be applied to the phenotype directly, avoiding a requirement for genetic representation.

A method of determining and maintaining population size. The basic evolutionary algorithm uses a single population with a constant number of individuals for each generation. Other strategies allow a population to gradually change via a steady-state mechanism, or allow the population to grow and decay based on some measure of external resources.

A termination criterion. Typically this is based on a performance measure (i.e. a minimum desired fitness measure) or a total effort in terms of number of generations, clock time, or computer processing time.

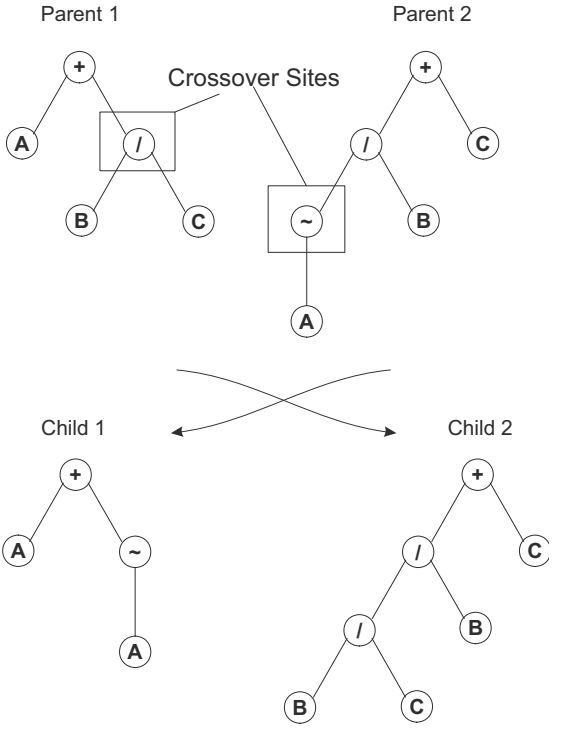


Figure 5.1. Genetic operators of crossover and mutation for bit-strings (GA).

There are a large variety of different strategies for each of these stages, however the underlying concepts are similar. A typical evolutionary algorithm with a single evolving population performs the following steps, based on the above components:

Commence with a randomly (or biased) population of N individuals, P(0).
$t := 0$
WHILE termination criterion not reached DO
Calculate the fitness F(n) for each individual $n \in$ P(t).
Repeat steps i…iv until N new individuals have been created in P(t+1):
Select a pair of individuals $n_1, n_2 \in$ P(t), using a selection method.
Based on a probability p_c, crossover n_1 and n_2 by taking a part of each individual and combining them to form a new individual n_1' and n_2'.
Based on a probability p_m, mutate n_1' and n_2'.
Insert n_1' and n_2' into P(t+1).
$t := t + 1$

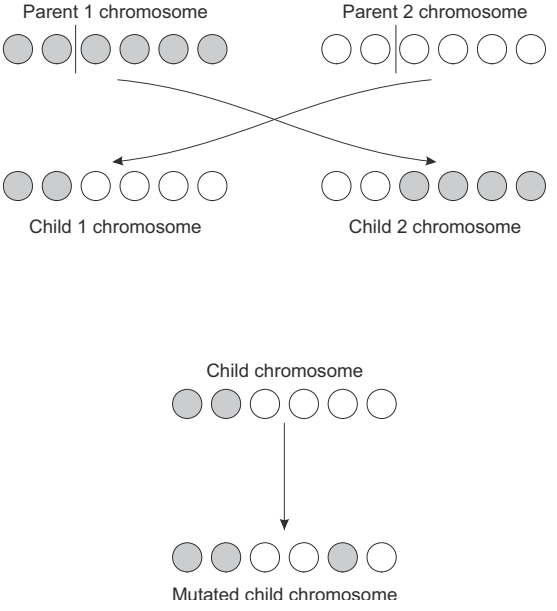


Figure 5.2. Crossover using tree structures (GP)

The details of the variation operators depend largely on the structure used to represent the individuals of the population. The choice of an appropriate representation is a key ingredient in the development of an evolutionary algorithm, however current theory (Wolpert and Macready 1997) suggests that no one representation or set of variation operators is most useful over all fitness functions. Therefore, the user is required to tailor the representation and variation operator to the problem at hand.

Figure 5.1 shows typical variation operators (crossover and point mutation) for individuals represented as bit strings. The site(s) for crossover is commonly selected at random and the parent material is recombined to create two new offspring. Mutation is normally applied with a certain probability to each site of the bit string, typically transforming the value from a 0 to 1, or vice versa. Figure 5.2 shows crossover using a tree structure as the individual representation (such as found in Genetic programming (GP)). Commonly, two random crossover sites are selected from the parents, and the subtrees below this site are swapped to create two new children. Mutation of a tree structure involves randomly selecting an internal node, deleting the subtree below this node and randomly creating a new subtree. The size of the new subtree is limited to some maximum depth of tree to limit the individual size. This can also be considered as a means of limiting the specialization of the tree representation: a smaller tree typically represents a more

general (parsimonious) solution. It is important to realize that the representations and variations offered above are merely two of many potential representations and the choice of the most appropriate method is left to the user to determine for each problem. This is an important point to address for the researcher as different representations may affect the utility of the evolutionary approach on the search space in question.

5.3.2 Summary

Evolutionary algorithms use concepts based on biological evolution to evolve solutions to problems for optimization, simulation and models of individual, group and community interaction. These approaches allow useful solutions to nonlinear problems to be generated in real-time and allow complex systems to be developed and described that are difficult to model with standard mathematical approaches.

5.4 Ecological Modelling and Evolutionary Algorithms

Section 5.2 discussed various goals of ecological modelling and the traditional frameworks used to understand and represent ecological problems. The following sections will describe various applications of EAs to develop or supplement models that have become standard approaches to understanding the patterns in ecology. Areas of research will also be highlighted that can extend current ecological theory based on evolutionary techniques.

5.4.1 Equation Discovery

Developing equations to describe the functional relationship between variables in a system is a key goal to understanding ecological systems. Typically, a differential or difference equation is created to describe the system, constants of the equations are tuned and the equations used to model the system. Given a set of data describing the independent and dependent variables, the goal is to produce an equation that models this information. Ideally the form of the solution should be constrained to allow only physically meaningful interpretations to be produced. One such example of an evolutionary system that allows this to occur is based on GP (Whigham 1995; Whigham and Crapper 1999), and uses a context-free grammar to allow bias in the form of evolved solutions. This approach has been successfully applied to freshwater system modelling, by creating equations that

predict the concentration of chlorophyll-*a* (Whigham and Recknagel 1999) and allows the equations to express relationships between variables as a function of past values, average values and nonlinear mathematical functions such as exponential, logarithms and power functions. The expressive nature of the equations allows more detailed exploration of the patterns then could be achieved using a statistical approach, since the constraints of independence and linearity are not required.

5.4.2 Optimisation of Difference Equations

There have been many developments of differential and difference equations to predict ecological response. Often there are difficulties in tuning the parameters of these equations; since the equations respond in a nonlinear fashion and so simple hill climbing search algorithms (i.e. dynamic programming) do not perform adequately. Since these problems can be framed as an optimization of the parameters of the difference equation, evolutionary algorithms are a suitable approach. One such example has been to use a GA to tune the parameters of a difference equation, with the parameters constrained within known physical limits. Each candidate solution (population member) represented a vector of the parameter values, and the fitness function was a measure of how well the difference equation predicted the measured data describing the freshwater system (Whigham and Recknagel 1999). Constraining the parameter values and using independent training and test data sets the equation parameters were evolved to produce far greater accuracy and generalization ability (the RMSE for the unseen test period of 1986 and 1993 was originally 91.46 and reduced to 46.75, as shown in Figure 5.3). This gave the evolved difference equation accuracy that was comparable to neural network and GP applications for the same data set.

Extensions to this work investigated evolving components of the difference equation to derive new terms (such as the grazing term) to substitute in the equation. This allowed an exploration of other forms of representation for this term, and concluded by demonstrating that the grazing term is likely to not be a linear function of chlorophyll-*a* concentration (Whigham and Recknagel 2000). This work was also extended to demonstrate that a complete differential equation could be evolved, however as the degrees of freedom increased, the possibility of exploiting other relationships in the data, and therefore not producing a physically based solution, became more likely.

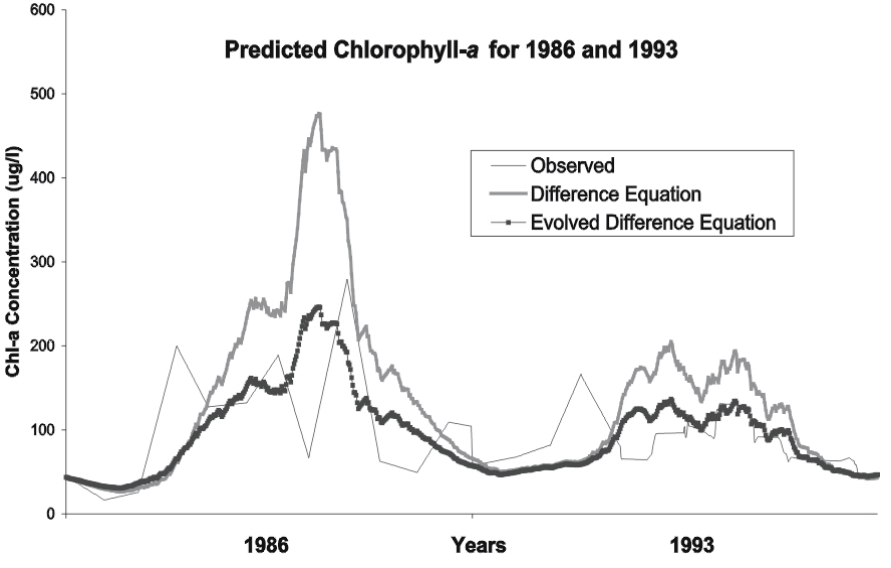


Figure 5.3. Evolved difference equation versus original parameter settings for the two unseen test years: 1986 and 1993.

5.4.3 Evolving Differential Equations

A common approach in ecology is to produce a set of differential equations to represent the relationships between variables in a system. Previous work has shown that searching for suitable differential equations based on ecological data are possible (Todorovski et al. 1998). In this work, *Lagramge*, an equation discovery system, was used to define the space of possible model structures and to automate the modelling of phytoplankton growth. Although this approach did not use an evolutionary system to perform the search (the search was a systematic breadth-first search of the possible equations) there is clearly an opportunity to extend the work using evolution to examine a larger search space of possible equations. Work that demonstrates the use of evolving differential equations is that of Sakamoto and Iba (Sakamoto and Iba 2001). In this work, equations representing a number of coupled differential equations were evolved using GP, however this approach has not currently been applied to ecological data.

5.4.4 Rule Discovery

Expressing knowledge or models in the form of rules became popular during the 1970's with the advent of expert systems (Buchanan and Shortliffe 1984). There has been a long history of rule discovery using evolutionary techniques (Wilson 1987; De Jong 1988; Grefenstette 1988; Robertson and Riolo 1988; De Jong and Spears 1991; Frey and Slate 1991; Lees and Ritman 1991; Corcoran and Sen 1994; Sipper 1994). The attraction with rules is that they can be easily interpreted by a user, and can be treated independently of the system in which they were created. A recent approach to rule discovery using evolution has been described by Bobbin and Recknagel (Bobbin and Recknagel 2001), where a set of rules for predicting various algal concentrations were created. For example, the following rule set described *Microcystis* dynamics (Recknagel et al. 2002):

IF P >= 126 AND P <= 81.7 µg/l
THEN *Microcystis* = 500,000 cells
ELSE
IF pH >= 9.72 THEN *Microcystis* = 500,000 cells
EXCEPT IF T <= 19.5 ºC AND T >= 5.67 ºC THEN
Microcystis = 3,000 cells
ELSE
IF N/P >= 47.2 AND N/P <= 55.2 THEN
Microcystis = 0
ELSE
IF S >= 95.5 cm THEN *Microcystis* = 3.5 cells
EXCEPT IF T >= 5.67 ºC AND T <= 15.7 ºC THEN
Microcystis = 0
OR IF N >= 1,110 µg/l THEN *Microcystis* = 0
IF P >= 15.6 µg/l AND P <= 116 µg/l
THEN *Microcystis* = 100,000 cells
EXCEPT IF T >= 26.7 ºC THEN *Microcystis* = 500,000 cells
EXCEPT IF S <= 160cm then *Microcystis* = 100,000 cells
ELSE
IF N >= 757 µg/l AND N <= 1,690 µg/l THEN
Microcystis = 0 cells
EXCEPT IF T >= 15.7 ºC AND T <= 26.7 ºC THEN
Microcystis = 3,000 cells
EXCEPT IF P <= 15.6 µg/l THEN
Microcystis = 0 cells
ELSE
IF T >= 15.7 AND T <= 26.7 ºC THEN
Microcystis = 100,000 cells
EXCEPT IF S <= 160 cm AND S >= 74.4 cm THEN
Microcystis = 0 cells

ELSE
IF T >= 5.67 ºC AND T <= 15.7 ºC THEN
Microcystis = 3,000 cells
ELSE
Microcystis = 100,000 cells.

This model successfully predicted both the timing and magnitude of *Microcystis* on unseen data from the same lake environment and was assessed as having a physically plausible interpretation.

A second example of rule discovery involved evolving a set of rules to predict the habitat density of a spatially-distributed marsupial (Whigham 2000). This work used a grammar-based GP system to show that spatially-explicit rules could be evolved that predicted the location and density of a marsupial, based on surveyed information and a set of spatial data. The resulting model was used to question the currently accepted home range for these marsupials. The evolved rules allowed expressions to be constructed that could not be easily formed using other techniques (McKay et al. 1997).

Several other rule-based systems, based on using a GA, have been created. *GARP* (Stockwell and Peters 1999) allowed the explicit integration of geographical data and a GA to discover rule sets that expressed spatial knowledge. This system has been used to automate the predictive spatial modelling of the distribution of species of plants and animals. *BEAGLE* (Fox et al. 1994), a GA for constructing logical expressions, was used to generate rules that could predict presence/absence of the duck species *Aythya ferina* on gravel pit lakes in southern Britain. *GAFFER* (Jeffers 1999) allowed the discovery of rules for numerical prediction and classification, and was designed to apply to real data sets where little background knowledge of relationships between variables was available. *GAFFER* was applied to discover rules that characterised habitat features determining the abundance of individual aquatic species.

5.4.5
Modelling Individual and Cooperative Behaviour

Modelling and understanding individual behaviour in an environment is of fundamental ecological interest. Work has been done using GP to evolve foraging strategies based on a model of the *Anolis* lizard (Koza et al. 1992). This work aimed at understanding what was an optimal foraging strategy, based on the abundance of insects, the velocity of the lizard, and the spatial relationship between the lizard and insects. The work demonstrated that for a set of insect abundance, lizard velocities and spatial placement the system evolved a sequence of progressively improved strategies. The models were expressed as a mathematical function that included decision points, based on an if-then function, and the current observed spatial positions of the insects, as viewed by the lizard.

These types of models show promise in producing hypotheses regarding individual behaviour and proposing theories that could be tested in the field.

A second example of individual behaviour is the study of trail following, in particular ant foraging and the use of pheromones for trail marking. The simplest concept is the evolution of trail following, as demonstrated by the Sante Fe trail experiments (Koza 1992). Here an artificial ant must learn to produce a trail following strategy that discovers food placed along a trail scattered amongst a 144-cell grid. GP was shown to be capable of finding solutions to this problem, even though the fitness function was based purely on the number of food units discovered within a certain number of evaluations. This type of model indicates that complex spatial behaviour for an individual can be evolved and studied, which may be used to understand observed foraging behaviour of real species.

Extending the concepts of a single population attempting to find a single best solution, coevolutionary algorithms use a number of subpopulations where each subpopulation evolves competing (rather than cooperating) solutions (Cohoon et al. 1987; Whitley and Starkweather 1990). Extensions of these ideas include the cooperative coevolutionary genetic algorithm (GA) (Potter and De Jong 1994), where a subpopulation represents a species that solves one particular aspect of a problem. The final complete solution is obtained by assembling representative members from each subpopulation. The subpopulations evolve independently, using a GA, where the goal is to have each subpopulation solve one aspect of the problem. This has similarities to the concepts of speciation, where each subpopulation finds a niche in the solution space to exploit. Other approaches with coevolution allow the modelling of individual behaviour in the population to produce models of predator-prey interactions (Cliff and Miller 1996; Haynes and Sen 1996; Rosin and Belew 1997) and other forms of competition. Each individual is represented as a bit string, neural network, or symbolic function that can evolve to produce behavior based on the competition produced from other individuals in the population. These approaches have been successful in demonstrating concepts such as diversity, extinction and genetic drift. However, these systems are difficult to interpret when applied to real ecosystem behavior or when they are coupled with data based on measured systems.

Extensions to multiple populations use the explicit representation of space, using CA or other grid-based representations, to allow spatial interactions to be explicitly represented. An early example of this approach was ‘Tierra’ (Ray 1992). Here organisms are represented as simple computer programs that compete for the memory and processing resources of the computer. The population changed over time through reproduction with mutation, and was constrained in terms of total size by having old individuals, or those that performed poorly, being gradually removed. The work showed that parasites could evolve in the population that used parts of other organisms for their own benefit.

A more complex system using self-replication and cooperating bits of computer code that evolved in a virtual computer world was called ‘Avida’ (Adami 1998). This was an extension of ‘Tierra’, with spatial structure based on a grid. The

results of this work demonstrated that evolving individuals with spatial structure showed some of the observed properties from real systems, such as power laws and self-organised criticality (Bak 1996). However, it should be noted that although a model shows self-organized criticality, or some other property that can be interpreted as ecological based, it is not necessarily useful as a model of an ecosystem. Few works based on these complex system approaches have been able to produce theoretical predictions that have been testable with real ecosystems (however, see Section 5.4.6 for one example).

Extending this concept to communities, GP has been used to demonstrate the evolution of cooperative behavior (Koza 1994). This work showed that high-level cooperative behavior of a community of ants, operating in parallel and with only local sensing, emerges by evolving each individual. Other work has used the behavior of real ants to construct systems that solve optimization problems (Dorigo and Caro 1999). These works relate to community ecology, where the structure and function of a community can be shown to evolve based on a simple task.

Simulated evolution has also been used to study the interactions of coevolving individuals within a population (for example (Kaufmann and Johnson 1992; Angeline and Pollack 1993; Fogel 1993; Ashlock et al. 1996)). Allowing the fitness function to depend on the constituents of the population, rather then being a fixed measure against a problem, causes the population to coevolve. Many of these simulations are based on an idealized model of interaction referred to as the interated prisoner's dilemma (IPD). IPD describes the interactions between individuals in a competitive environment, where there are varying payoffs based on whether two individuals cooperate, defect or some combination. IPD has been successful in modelling the evolution of cooperative strategies between individuals in a community, and is a useful theoretical model of social interaction. For example, Ashlock et al. (1996) studied partner selection as a process in social interactions. The models showed that, based on the degree to which individuals are intolerant of defections and social isolation, various ecologies dominated. However, like many theoretical studies, this work has not been extended to predictions of real ecosystem structure, although there is clearly an opportunity to use this form of modelling for predictions of population structure, cooperative behavior and more complex interactions such as language development and symbiosis.

Extending this concept to community assemblages has been achieved using the Echo model (Forrest and Jones 1994; Hraber and Milne 1997), where a set of agents coevolves under the pressure of invasion and agent interaction. This work allowed a study of species abundance patterns, community assembly rules, species richness and ecosystem stability. Other work using a genetic approach has allowed a model of plant-herbivore interactions (Hartviggsen and Starmer 1995). In this work the plants have simulated genes that infer a certain resistance to grazing, and the herbivores have simulated genes that produce conditions that overcome these plant defenses. The work allowed an investigation of density

dependence and allowed several conclusions relating to coevolutionary patterns to be inferred.

5.4.6 Predator-Prey Algorithms

Following the traditional predator-prey models studied by Lotka and Volterra (Lotka 1927) a number of works have studied competitive coevolution to model predator-prey behaviour (Haynes et al. 1995; Cliff and Miller 1996; Haynes and Sen 1996; Rosin and Belew 1997). These models use competition between evolving communities of predators and prey to demonstrate how survival strategies and behaviour can coevolve. Since the complexity of real pursuit-evasion are too difficult to code as a simple set of differential equations, the use of evolving models affords more complex instances, such as perceptual specialization, behaviour prediction and planning, to be studied.

The concept of evolutionary stable strategies (ESS) (Smith and Price 1973; Smith 1982) has been commonly used to predict the behaviour and characteristics of naturally evolved organisms (Dawkins 1989; Motro 1991; Visser et al. 1992; Wolf and Waltz 1993). The behaviour of complex adaptive systems are anticipated by examining an evolutionary game with various possible strategies for each player and prescribed payoffs dependent on the play of all participants. The equilibrium conditions of the game are determined mathematically and it is assumed that once the players' strategies have reached an equilibrium, they will tend to remain in that condition, barring external influences. The hawk-dove game is a typical example of a game that can lead to an ESS condition given a variety of assumptions regarding the population including an infinite population and payoffs to competitors described only on the average. With these assumptions, mathematics can be used to determine the ESS for the population. Evolutionary computation has been applied to the hawk-dove game in order to determine if the ESS maintains value under the realization that in natural populations, the assumptions mentioned above are not realistic (Fogel and Fogel 1995; Fogel and Fogel 1997; Fogel et al. 1998). Under more realistic conditions of finite populations and stochastic payoffs, the evolutionary simulations demonstrated that populations may evolve in trajectories that are unrelated to an ESS, even in very simple systems with small populations. This more realistic modelling of an evolving system has therefore cast doubt on the utility of ESSs to provide useful explanations of the behaviour of populations even at relatively low levels of selection, even under persistent mixing.

5.4.7 Modelling Hierarchical Ecosystems

Living systems involve many levels of hierarchical interaction, from the genetic level through the individual to the community and complete ecosystem (Conrad and Pattee 1970). Conrad and Pattee argued that a theory of evolution that does not reflect this structure cannot be expected to be useful in terms of predictions and models of real systems. They presented a model (EVOLVE I) based on a population of cell-like organisms subject to a strict mass conservation law. This limitation of resources induced competitive behavior between individuals in the population. The system showed that artificial life models could lead to discoveries about biological evolution.

Extensions of this work involved the construction of three nested models, each corresponding to a different layer of biological organization (Rizki and Conrad 1985; Conrad and Rizki 1989; O'Callaghan and Conrad 1992). EVOLVE III (O'Callaghan and Conrad 1992) included organizations for genetic structure, organisms and populations, where each of these components was modelled independently. The genetic structure contained simple representations of DNA and an algorithm for abstracting transcription, translation and protein folding. Organisms had a number of phenotypic traits, including response to light intensity, rates of energy usage, protection and aggression mechanisms and a life cycle history. Each trait was coded by a collection of genes that could be mutated during reproduction to allow variation in future organisms. The ecosystem was represented as a set of populations and an abiotic environment, where each population was composed of individual organisms. The system was designed to allow various types of populations, organisms and genetic structure to be independently studied. For example, a simple population consisted of producers and decomposers. Producer organisms used nutrients from the environment and returned them to the environment with a degraded energy value. In turn, decomposers used these degraded nutrients and returned them to the environment as nutrients available to producers after a period of time. This allowed the system to produce a food cycle where mass was conserved.

Demonstrating the link between theoretical and real ecosystems, EVOLVE III was used to explore relationships between adaptability of populations and the variability of the environment. Results from the theoretical model suggested that populations cultured in a constant environment usually dominated those cultured in a variable environment when both were placed in a variable environment at an early stage of development. This pattern was verified by laboratory experiments and indicates the potential predictive value of the model. This work represented a significant approach and goal of simulation studies using evolution: the systems must be able to demonstrate behavior that can be used to interpret real ecosystems and that the results should be verifiable through laboratory experiments. EVOLVE IV (Brewster and Conrad 1998) was designed to explore the effects of environmental uncertainty on niche proliferation and the evolution of interspecific

interactions. Although many models of ecosystems have been built in the past decade, especially under the field of artificial life, the conclusions from these models have been difficult to interpret back to real systems and give predictions that could be verified experimentally.

More recently there has been interest in the use of exergy and other thermodynamic measures (Jorgensen 1992; Jorgensen 1992; Jorgensen et al. 1995; Salomonsen and Jensen 1996; Svierezhev 2000) to give global descriptions of system dynamics. Incorporating these concepts into evolutionary ecosystem models would be a positive direction for future research.

5.5 Conclusion

The previous sections have described some of the basic applications of evolutionary computation techniques to various aspects of ecological modelling. Although there are many areas that have not been given adequate attention, it is clear that the use of difference and differential equations, the modelling of cooperation and community structure, the use of space and spatial behavior and the construction of hierarchical organization are areas where evolutionary computation techniques match well with ecological modelling. Models from large-scale behavior of communities, through to the way in which genetic material evolves in a species, can be studied using these types of models. The future is extremely positive for these evolutionary techniques to support and extend the current understanding of ecological processes and functions.

References

Adami C (1998) Introduction to Artificial Life. New York, Springer-Verlag

Angeline PJ, Pollack JB (1993) Competitive Environments Evolve Better Solutions for Complex Tasks. Proceedings of the 5th International Conference on Genetic Algorithms. S. Forrest, San Mateo, CA. Morgan Kaufmann: 264-270

Ashlock D, Smucker M D (1996) Preferential partner selection in an evolutionary study of prisoner's dilemma. Biosystems 37: 99-125

Bagley J (1967) The Behavior of Adaptive Systems Which Employ Genetic and Correlation Algorithms, Univ. Michigan, Ann Arbor

Bak P (1996) How Nature Works: The Science of Self-Organized Criticality. New York, Springer-Verlag

Barricelli NA (1954). Esempi Numerici di Processi di Evoluzione. Methodos: 45-68

Bobbin J, Recknagel F (2001) Knowledge Discovery for Prediction and Explanation of Blue-Green Algal Dynamics in Lakes by Evolutionary Algorithms. Ecol. Modelling 146, 1-3, 253-264

Bremermann H (1962) Optimization through Evolution and Recombination. Self-organizing Systems. M.C.Yovits, G. T. Jacobi and G. D. Goldstine. Washington DC, Spartan Books: 93-106

Bremermann H, Rogson M (1966) Global Properties of Evolution Processes. Natural Automata and Useful Simulations. H. H. Pattee, E. Edlasck, L. Fein and A. Callahan. Washington DC, Spartan Books: 3-41

Brewster J, Conrad M (1998) Evolve IV: A Metabolically-Based Artificial Ecosystem Model. *In:* Evolutionary Programming VII: 7th International Conference, EP98. V. W. Porto, N. Saravanan, D. Waagen, and A.E. Eiben (eds.), Springer-Verlag, New York: 473-492

Buchanan B, Shortliffe E Eds. (1984) Rule-Based Expert Systems, Addison-Wesley

Carter RN, Prince SD (1981) Epidemic models used to explain biogeographical distribution limits. Nature 293: 644-645

Cliff D, Miller GF (1996) Co-evolution of Pursuit and Evasion: Simulation Methods and Results. From animals to animats 4. P. Maes, M. Mataric, J. Meyer, J. Pollack and S. Wilson. Cambridge, MA, MIT Press: 506-515

Cohoon JP, Hegde SU (1987) Punctuated equilibria: A parallel genetic algorithm. Proceedings of the Second International Conference on Genetic Algorithms, Lawrence Erlbaum Associates

Colasanti RL, Grime JP (1993) Resource dynamics and vegetation processes: a deterministic model using two dimensional cellular automata. Functional Ecology 7: 169-176

Comins HN, Hassell MP (1992) The spatial dynamics of host-parasitoid systems. Journal of Animal Ecology 61: 735-748

Conrad M, Pattee HH (1970) Evolution Experiments with an Artificial Ecosystem. J. Theor. Biol. 28: 393-409

Conrad M, Rizki M (1989) The artificial worlds approach to emergent evolution. Biosystems 23: 247-260

Corcoran AL, Sen S (1994) Using real-valued genetic algorithms to evolve rule sets for classification. Proceedings of the IEEE Conference on Evolutionary Computation

Dawkins R (1989) The Selfish Gene. Oxford, Oxford Univeristy Press

De Jong K (1988) Learning with genetic algorithms: An overview. Machine Learning 3(2,3): 121-138

De Jong K, Spears WM (1991) Learning Concept Classification Rules Using Genetic Algorithms. Proceedings of the Twelfth International Conference on Artificial Intelligence. 2: 651-657

Dorigo M, Caro GD (1999) The Ant Colony Optimization Meta-Heuristic. New Ideas in Optimization. D. Corne, M. Dorigo and F. Glover. London, McGraw-Hill: 11-32

Fogel D (1998) Evolutionary Computation: The Fossil Record, IEEE Press. Piscataway, New Jersey

Fogel D (2000) Evolutionary Computation: Toward a New Philosophy of Machine Intelligence, IEEE Press. PIscataway, New Jersey

Fogel DB (1993) Evolving behaviors in the iterated prisoner's dilemma. Evol. Comput. 1: 77-99

Fogel DB, Fogel GB (1995) Additionally, models that incorporate spatial and temporal information, and those based on populations of individuals, are difficult to formulate or express. Evolutionary Programming IV: Proceedings of the Fourth Annual Conference on Evolutionary Programming. J. R. McDonnell, R. G. Reynolds and D. B. Fogel, MIT Press, Cambridge, MA: 565-577

Fogel DB, Fogel GB (1997) On the instability of evolutionary stable strategies. BioSystems 44: 135-152

Fogel GB, Andrews PC (1998) On the instability of evolutionary stable strategies in small populations. Ecol. Modelling 109: 283-294

Fogel LJ (1962) Autonomous Automata. Industrial Research 4: 14-19

Fogel LJ, Owens AJ (1966) Aritificial Intelligence Through Simulated Evolution. New York, John Wiley

Forrest S, Jones T (1994) Modeling complex adaptive systems with Echo. Complex Systems: Mechanisms of Adaptation. R. J. Stonier and X. H. Yu. Amsterdam, IOS Press: 3-21

Fox AD, Jones TA (1994) Food supply and the effects of recreational disturbance on the abundance and distribution of wintering Pochard on a gravel pit complex in southern Britain. Hydrobiologia 279/280: 253-261

Fraser A (1957) Simulation of Genetic Systems by Automatic Digital Computers I. Introduction. Australian J. of Biol. Sci. 10: 484-491

Fraser A (1962) Optimization through Evolution and Recombination. Self-organizing Systems. M.C.Yovits, G. T. Jacobi and G. D. Goldstine. Washington DC, Spartan Books: 93-106

Frey PW, Slate DJ (1991) Letter recognition using Holland-style adaptive classifiers. Machine Learning 6(2): 161-182

Friedberg RM (1958) A Learning Machine: Part I. IBM J. of Research and Development 2: 2-13

Friedberg RM (1959) A Learning Machine: Part II. IBM J. of Research and Development 3(3): 282-287

Gillman M, Hails R (1997) An Introduction to Ecological Modelling: Putting Practice into Theory. Oxford, Blackwell Science Ltd

Grefenstette JJ (1988) Credit assignment in rule discovery systems based on genetic algorithms. Machine Learning 3(2,3): 225-245

Hartviggsen G, Starmer WT (1995) Plant-herbivore coevolution in a spatially and genetically explicit model. Artificial Life 2: 239-258

Hassel MP, Comins HN (1976) Discrete time models for two-species competition. Theoretical Population Biology 9: 202-221

Haynes T, Sen S (1996) Evolving Behaviour Strategies in Predators and Prey. Adaption and Learning in Multiagent Systems. G. Weis and S. Sen. Berlin, Springer Verlag

Haynes T, Wainwright R (1995) Strongly Typed Genetic Programming in Evolving Cooperation Strategies. Proceedings of the Sixth International conference on Genetic Algorithms. L. J. Eshelman: 271-278

Holland JH (1969) Adaptive Plans Optimal for Payoff-Only Environments. Proc. of the 2nd Hawaii Int. Conf. on System Sciences, Hawaii

Holland JH (1973) Genetic Algorithms and the Optimal Allocations of Trials. SIAM J. Comp 2(2): 88-105

Hraber P, Milne B (1997) Community Assembly in a model Ecosystem. Ecological Modelling 103: 267-285

Jeffers J (1999) Genetic Algorithms I. Machine Learning Methods for Ecological Applications. A. Fielding, Kluwer Academic Publishers: 107-121

Jorgensen SE (1992) Development of models able to account for changes in species composition. Ecological Modelling 62: 195-208

Jorgensen SE (1992) Exergy and ecology. Ecological Modelling 63: 185-214.

Jorgensen SE, Nielsen S (1995) Emergy, envirton, exergy and ecological modelling. Ecological Modelling 77: 99-109

Judson OP (1994) The rise of the individual-based model in ecology. Trends in Ecology and Evolution 9: 9-14

Kaufmann SA, Johnson S (1992) Coevolution to the edge of chaos: coupled fitness landscapes, poised states and coevolutionary avalanches. Artificial Life II. C. G. Langton, C. Taylor, J. D. Farmer and S. Rasmussen, Reading, MA:Addison-Wesley: 325-370

Koza JR, Rice JP (1992) Evolution of food foraging strategies for the Caribbean anolis lizard using Genetic Programming. Adaptive Behavior 1(2): 47-74

Koza JR (1992) Genetic Programming:On the Programming of Computers by means of Natural Selection, Cambridge, Mass.:MIT Press

Koza JR (1994) Evolution of emergent cooperative behavior using genetic programming. Computing with Biological Metaphors. R. Paton. London, UK, Chapman & Hall: 280-297

Lees B, Ritman K (1991) Decision-tree and rule-induction approach to integration of remotely sensed and GIS data in mapping vegetation in disturbed or hilly environments. Environmental Management 15(6): 823-831

Levins R, Culver D (1971) Regional coexistence of species and competition between rare species. Bulletin of the Entomological Society of America 15: 237-240

Lotka AJ (1927) Fluctuations in the abundance of species considered mathematically (with comment by V. Volterra). Nature 119: 12-13

McCauley DE (1995) Effects of population dynamics on genetics in mosaic landscapes. Mosaic Landscapes and Ecological Processes. L. Hansson, L.Fahrig and G. Merriam. London, Chapman and Hall: 178-198

McKay RI, Pearson RA (1997) Learning Spatial Relationships: Some Approaches. GeoComputation '97, University of Otago, Dunedin, New Zealand

Motro U (1991) Co-operation and defection: playing the field and the ESS. J. Theor. Biol. 151: 145-154

O'Callaghan J, Conrad M (1992) Symbiotic interactions in the EVOLVE III ecosystem model. Biosystems 26: 199-209

Potter MA, De Jong K (1994) A Cooperative Coevolutionary Approach to Function Optimization. Lecture Notes in Computer Science 866: 249-258

Ray TS (1992) An apporach to the synthesis of life. Artificial Life II, Reading, MA, Addison Wesley

Rechenberg I (1965) Cybernetic Solution Path of an Experimental Problem, Royal Aircraft Establishment, Library Translation

Recknagel F, Bobbin J, Whigham P, Wilson H (2002) Comparative application of artificial neural networks and genetic algorithms for multivariate time series modelling of algal blooms in freshwater lakes. Journal of Hydroinformatics 4, 2, 125-133.

Rizki M, Conrad M (1985) Evolve III: A discrete events model of an evolutionary ecosystem. BioSystems 18: 121-133

Robertson GG, Riolo RL (1988) A tale of two classifier systems. Machine Learning 3(2,3): 139-159

Rosenberg R (1967) Simulation of Genetic Populations with Biochemical Properties, Univ. Michigan, Ann Arbor

Rosin CD, Belew RK (1997) New Methods for Competitive Coevolution. Evolutionary Computation 5(1): 1-29

Sakamoto E, Iba H (2001) Inferring a System of Differential Equations for a Gene Regulatory Network by using Genetic Programming. IEEE Congress on Evolutionary Computation, Seoul, Korea, IEEE Piscataway, NJ

Salomonsen J, Jensen J (1996) Use of a lake model to examine exergy response to changes in phytoplankton growth parameters and species composition. Ecological Modelling 87: 41-49

Schwefel HP (1965) Kybernetische Evolution als Strategie der Experimentellen Forschung in der Strömungstechnik, Technical University of Berlin

Silverton J, Holtier S (1992) Cellular automation models of interspecific competition for space - the effect of pattern on process. Journal of Ecology 80: 527-534

Sipper M (1994) Non-Uniform Cellular Automata: Evolution in Rule Space and Formation of Complex Structures. Artificial Life IV, MIT Press

Smith JM (1982) Evolution and the Theory of Games. Cambridge, Cambridge University Press

Smith JM, Price GR (1973) The logic of animal conflict. Nature 246: 15-18

Stockwell D, Peters D (1999) The GARP modelling system:problems and solutions to automated spatial prediction. International Journal of Geographical Information Science 13(2): 143-158

Svierezhev Y (2000) Thermodynamics and ecology. Ecological Modelling 132: 11-22

Todorovski L, Dzeroski S (1998) Modelling and prediction of phytoplankton growth with equation discovery. Ecological Modelling 113: 71-81

Visser ME, Alphen JJMV (1992) Adaptive superparasitism and patch time allocation in solitary parasitoids: an ESS model. J. Anim. Ecol. 61: 93-101

Watkinson AR (1987) Plant Population Dynamics. Plant Ecology. M. J. Crawley. Oxford, Blackwell Scientific Publications: 137-184

Whigham PA (1995) Inductive Bias and Genetic Programming. Genetic Algorithms in Engineering Systems: Innovations and Applications (GALESIA '95): 461-467

Whigham PA (2000) Induction of a marsupial density model using genetic programming and spatial relationships. Ecological Modelling 131(2-3): 299-317

Whigham PA, Crapper PF (1999) Time series modelling using genetic programming: An application to rainfall-runoff models. Advances in Genetic Programming 3. L. Spector,

W. B. Langdon, U. O'Reilly and P. J. Angeline, MIT Press, Cambridge, MA, USA. 5: 89-104

Whigham PA, Recknagel F (2000) Evolving Difference Equations to Model Freshwater Phytoplankton. 2000 Congress on Evolutionary Computation, San Diego, USA, IEEE, Piscataway, NJ

Whigham PA, Recknagel F (1999) Predictive Modelling of Plankton Dynamics in Freshwater Lakes using Genetic Programming. MODSIM '99 International Congress on Modelling and Simulation, Hamilton, New Zealand, The Modelling and Simulation Society of Australia and New Zealand Inc

Whitley D, Starkweather T (1990) Genitor II: a distributed genetic algorithm. Journal of Experimental and Theoretical Artificial Intelligence 2: 189-214

Wilson SW (1987) Classifier systems and the animat problem. Machine Learning 2(3): 199-228

Wolf LL, Waltz EC (1993) Alternative mating tactics in male white-faces dragonflies: experimental evidence for a behavioural assessment ESS. Anim. Behav. 46: 325-334

Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation: 1:67-82

Chapter 6

Ecological Applications of Adaptive Agents

F. Recknagel

6.1 Introduction

Ecologists are constantly searching for new modelling paradigms in order to simulate realistically the distinct nature of ecosystems by computer models. The ecosystem concept as established by Forbes (1887) had the most forming influence on ecosystem modelling in the past century. It no longer bears close examination as ecosystems like lakes are known to evolve and being driven by exogenous forces rather than existing permanently and in isolation. However, the ecosystem approach resulted in valuable databases from monitoring as well as quantitative and qualitative descriptions of ecosystem dynamics and has made ecology a predictive science (Rigler and Peters 1995). Computer models resulting from the ecosystem concept were mainly based on differential equations (DE) for well-defined ecological entities and processes, adjusted by measured or estimated parameters. Radtke and Straskraba (1980) firstly tried to overcome the rigidity of such models by parameter optimization of ecological goal functions relevant to lake ecosystems as introduced by Straskraba (1977). The authors considered their results as contribution to a structural self-optimising ecosystem model but admitted that more adequate models and more suitable optimisation procedures would be needed to make it a success. In order to overcome model rigidity, Kaluzny and Swartzman (1985) suggested a library of alternative representations of ecological processes from where a simulation model picks the most relevant one for a specific ecological situation. The authors concluded that their approach was limited by validation data and 'the difficulty of tracing model response to single processes' (Kaluzny and Swartzman 1985). Jorgensen and Mejer (1979) introduced the thermodynamic entity exergy for holistic ecosystem modelling that has led to the concept of structural dynamic models (Jorgensen 1986). It equips an ecosystem model with a global rather than local goal function, namely maximizing exergy storage, to be satisfied by optimising process parameters in the course of simulation. Even though this approach avoids the problem of biasing by 'local' optima as faced by Radtke and Straskraba (1980), it may require more adequate models and more suitable optimisation procedures as well.

Machine learning techniques such as artificial neural networks (ANN) (Rumelhart et al. 1986) and evolutionary computation (EC) (Holland 1992) allow looking at the same problem from a different angle. They are inductive techniques

and allow extracting empirical patterns as reflected by multivariate nonlinear time series data. Even though the range and extent of data available may limit ANN, EC can explore both causal and empirical information by means of hybrid frameworks to induce and evolve models (Bobbin and Recknagel 2001; Whigham and Recknagel 2001). However predictive capacity of resulting models still relies on underlying causal and empirical knowledge. The application of adaptive agents (AA) (Holland 1992; Holland 1998) is an attempt to go one step further: to evolve ecosystem structures and behaviours by emerging, submerging, interacting and evolving ecological entities simulated by adaptive agents.

The present paper reviews current developments of individual-based AA for microbial and terrestrial ecosystems, and designs a concept how state variable-based AA can be applied in order to simulate evolving species abundance and succession in aquatic ecosystems. The proposed concept is currently developed and tested towards adaptive lake ecosystem simulation. It is expected to overcome constraints by the rigidity of traditional dynamic ecosystem models and enable to evolve ecosystem structures and behaviours.

6.2 Adaptive Agents Framework

Holland (1992) introduced Echo (Fig. 6.1.) as a generic simulator designed to explore interactions among large numbers of different adaptive agents (AA). It provides for the study of populations of evolving, reproducing agents distributed over a geography with different inputs of renewable resources at various sites. Each agent has simple capabilities – offence, defense, trading, mate selection – determined by a set of "chromosomes". Chromosomes in each agent are differentiated into two classes:

Tag chromosomes determine the agent's external phenotypic characteristics and distinguish: offence tag, defence tag and mating tag. Tags are displayed on the exterior of an agent and are analogous to signature groups of an antigen or the logo of an organisation. Condition chromosomes determine what kinds of interactions take place when agents encounter one another and distinguish: combat (competition), trading (mutualism) or mating (reproduction).

The fact that an agent's structure is completely defined by its chromosomes, which are just strings over the resource alphabet $\{a,b,c,d\}$, plays a critical role in its reproduction. An agent reproduces when it "collects" enough letters to make copies of its chromosomes. An agent can collect these letters through its interactions: combat, trade, or uptake from the environment. Each agent has a reservoir in which it stores collected letters until there are enough of them for reproduction to take place. Interactions between agents, when they come into contact are determined by a simple sequence of tests based on their tags and conditions. In the simplest model they first test for combat, then they test for trading and finally they test for mating as follows:

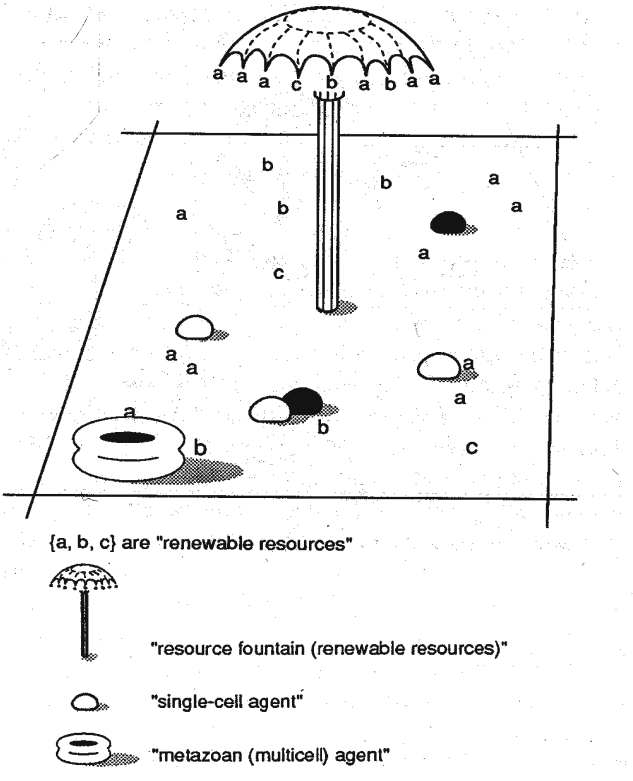


Fig. 6.1. Conceptual diagram of the adaptive agent model Echo (modified after Holland 1992)

Combat: Each agent checks its combat condition against the offence tag of the other agent. E.g., if the combat condition is given by the string *aad*, then this condition is matched by any offense tag that begins with the letters *aad*. If the combat condition of either agent matches the offense tag of the other, then combat is initiated. Combat can be initiated unilaterally by either agent. If combat is initiated the offence tag of the first agent is matched against the defense tag of the second and a score is calculated.

Trading: If combat does not take place, then the first agent in the pair checks its trading condition against the offence tag of the second agent, and vice versa. Unlike combat, which can be initiated unilaterally, trading is bilateral – a trade does not take place unless the trading conditions of both agents are satisfied. The trading condition in the simplest model has a single letter, as a suffix, that

specifies the resource being offered for trade. If the trade is executed then each agent transfers any excess of the offered resource (amounts over and above the requirements for its own reproduction) from its reservoir to the reservoir of its trading partner. Though this is a very simple rule, with no bidding between agents, it does lead to intricate, rational trading interactions as the system evolves. Trades that provide resources needed for reproduction increase the reproduction rate, assuring that agents with such rational trading conditions become common components of the population.

Mating: While an agent can reproduce asexually, simply making a copy of each of its chromosomes when it has accumulated enough resources (letters), there is also a provision for recombination of chromosomes. When agents come into contact and do not engage in combat, the mating condition of each agent is checked against the mating tag of the other. As with trade, mating is only executed as a bilateral action. Both agents must have their mating conditions satisfied for recombination to take place. If this happens, then the agents exchange some of their chromosome material, as with crossover under the genetic algorithm.

AA characterised by these simply defined capabilities provide for a rich set of variations illustrating the key kernel properties of complex adaptive systems. They were originally developed and applied for the study of complex adaptive economic systems (Holland and Miller 1991) such as stock markets (Wan and Hunter 1997) and businesses (Lin and Pai 2000). However modified versions of Echo have meanwhile been used to simulate spatial dynamics of species or populations represented by individuals strictly based on causal knowledge (Booth 1997; Schmitz and Booth 1997; Kreft, Booth and Wimpenny 1998). These examples are based on the assumption that local emergence or submergence of individuals is driven by interrelationships between well-defined individuals and their environment. Such an individual-based approach seems to be relevant to terrestrial ecosystems like forests (Schmitz and Booth 1997) where spatial spreading of individual tree species as an outcome of competitive success is of major interest. AA simulation of aquatic ecosystems requires a different approach as normally neither individual nor spatial aspects are relevant, nor are adequate data available.

6.3 Individual-Based Adaptive Agents

Individual-based modelling aims at naturally and easily simulating effects of complex ecological interactions such as individual variation, spatial processes, and cumulative stress. The concept was introduced by Huston, DeAngelis and Post (1988) who argued that in ecosystems “amplifying effects can arise from spatial non-uniformities and variations in the environmental conditions that each organism experiences, such as moisture and light for plant seedlings, or variable habitat and patchy distributions of preys for animals. Amplifying effects can also result from differences among individuals that are properties of the organisms

themselves such as size, age, physiological characteristics, and genetic variation". Even though the concept appeared to be plausible and applicable especially to the distinct spatial and heterogeneous nature of terrestrial ecosystems, Railsback (2001) identified two reasons why individual-based modelling has so far not been a 'very productive approach to ecology or ecological management': (1) the failure to encode models in software that allows the behaviour of the model's individuals to be observed and tested, (2) the use of inappropriate assumptions abound in individual-based models such as: using model components developed originally for one set of assumptions to simulate conditions under which those assumptions clearly are not met; applying relations and parameters developed for one spatial or temporal scale to other scales that they are not appropriate for; embedding empirical relations in models that are purported to be mechanistic; confusing individual- and population-level parameters. However Holland's paradigm on adaptive agents conceptualised in Echo (Fig. 6.1.) seems to reinforce individual-based modelling as it can be applied to spatially explicit simulation of individuals of species by single agents.

Gecko (Booth 1997) is an example of an individual-based adaptive agents simulation system implemented on the Echo framework. In order to simulate community effects of spatial competition, agents in Gecko are not constrained by lattice but extent and compete directly for space. In extension of Echo, Gecko simulates energetics explicitly. "The individual-based components of Gecko include cross primary production, over a constrained space, and several basic species types. Organisms are abstracted as spheres on the resource-producing plane. They have behaviours to acquire food, assimilate food at realistic efficiencies, and pay metabolic taxes at allometrically specified rates…Creatures interact locally, with their neighbourhoods circumscribed by their radii, determined in turn by their size – the biomass they have amassed. Thus in addition to having position in two dimensions, Gecko creatures have the spatiotemporal property of extent" (Booth 1997). Applications of Gecko to a hypothetical terrestrial food chain consisting of a plant, a herbivorous and a carnivorous animal revealed that the sequential implementation of seven basic rules for agent to agent interactions during each simulation step allowed to produce qualitatively sound results for different scenarios (Booth 1997; Schmitz and Booth 1997). Examples of basic rules considered in this study are: give everyone a chance to interact; distribute abiotic resources to autotrophic agents at the site; take maintenance tax; etc. In an attempt to simulate the growth of *Escherichia coli* from a single cell to a bacterial colony Gecko was implemented in a deterministic manner based on equations for cell growth kinetics, diffusion and colony expansion, and produced qualitatively sound results regarding colony structures for different glucose concentrations (Kreft, Booth and Wimpenny 1998).

Even though above described examples were rather simple and vague reflections of real ecosystems the authors bewailed the fact that their achievements were limited by causal knowledge from the individual to the ecosystem level – a limitation that is inherent to deductive approaches strictly based on causal knowledge.

6.4
State Variable-Based Adaptive Agents

Aquatic ecosystems such as lakes have a definite boundary with primary producers dominated by microscopic algal cells (1 to 200 µm) with generation times of hours to days, and secondary producers dominated by mesoscopic zooplankton (20 to 2000 µm) with generation times of days and weeks (Rigler and

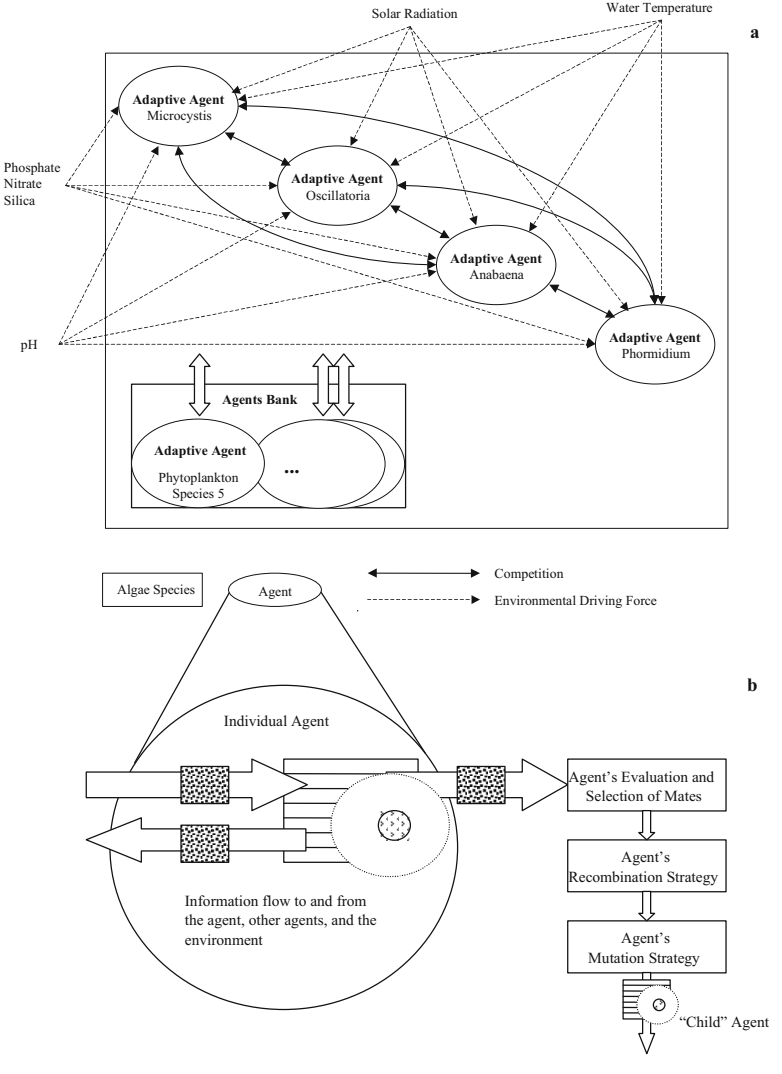


Fig. 6.2. Adaptive agents simulation of algal species dynamics

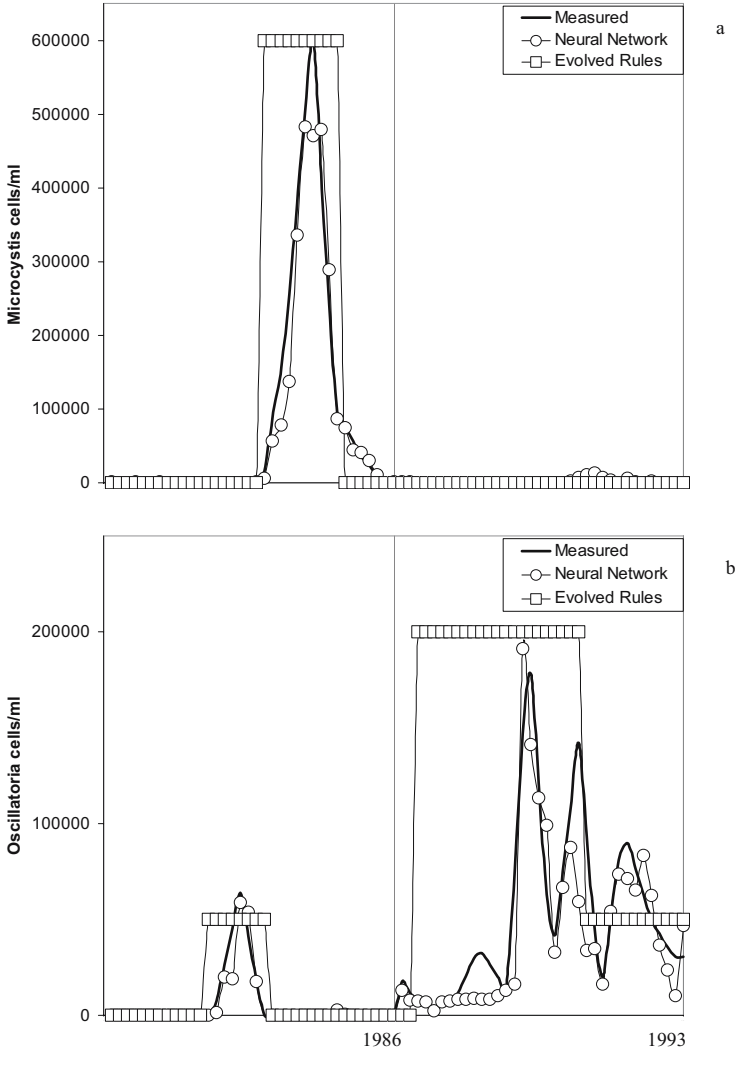


Fig. 6.3. Same-day-predictions of algal cells of *Microcystis* (a) and *Oscillatoria* (b) by artificial neural networks and evolved rules.

Peters 1995). Dissolved inorganic nutrients are homogeneously distributed within the euphotic surface layer where algal cells strongly interact and grow as a result of competition for nutrients and light. The wind continuously stirs the surface layer contributing to an almost homogeneous horizontal distribution of algal cells. Zooplankton may also be affected by wind but are mobile to a certain extent. They tend to form horizontal patches in response to food availability and predation pressure by fish. If we focus lake modelling on the euphotic zone as the scene of

primary and secondary production in lakes, we can imply that plankton communities are almost homogenously distributed and almost instantaneously responding to exogenous disturbances. As the prediction and explanation of instantaneous algal abundance and succession appear to be the biggest challenge to freshwater ecologists, the AA simulation of the spatial distribution of individuals as suggested for terrestrial ecosystems (e.g. Schmitz and Booth 1997) seems no longer relevant. Neither adequate knowledge nor data would be available to realistically reflect individual or spatial aspects of algae and zooplankton. By contrast the AA simulation of aquatic ecosystems needs to focus on the temporal distribution of plankton populations (respective functional groups) by means of state variable-based AA embodying evolutionary computation.

6.4.1 Algal Species Simulation by Adaptive Agents

Adaptive agents simulation of algal species dynamics is currently designed and developed according to Fig. 6.2. Four agents are considered initially to represent blue-green algae species typically competing in eutrophic freshwaters in summer: *Microcystis, Oscillatoria, Anabaena* and *Phormidium*. (see Fig. 6.2a). These four agents interact by competition and are determined by environmental driving forces such as solar radiation, water temperature, and nutrient loadings.

6.4.1.1 Embodiment of Evolutionary Computation in Agents

Each single agent is embodied by artificial Neural Networks (ANN), evolving differential equations (EDE) or evolving rules (ER) in order to maximise (adapt) their performance (abundance) in relation to current environmental conditions (nutrient loadings, light, temperature and abundance of competitors).

Case studies on ANN (Recknagel 1997; Recknagel et al. 1997), EDE (Whigham and Recknagel 2001) and ER (Bobbin and Recknagel 2001; Recknagel et al. 2002) have been conducted for the prediction of algal abundance and succession in lakes and reservoirs. Fig. 6.3. shows simulation results for *Microcystis* (a) and *Oscillatoria* (b) in Lake Kasumigaura predicted by ANN and ER. The underlying ER used for the same-day predictions in Fig. 6.3. are documented in Table 6.1.

Examples in Fig. 6.4. are based on 7-days-ahead predictions for chlorophyll-*a* and *Microcystis* in Lake Kasumigaura performed by EDE and ER (see Tab. 6.2.) and ANN (Recknagel et al. 2002), which were trained and extracted from the Lake Kasumigaura data base (Takamura et al. 1992). The underlying DE was adopted from the deterministic lake model SALMO (Recknagel and Benndorf 1982).

During the AA simulation of algal dynamics in a specific lake, each agent adapts steadily to occurring environmental conditions by producing the best adapted model or “offspring” agent based on its evaluation and selection of mates, recombination strategy and mutation strategy (see Fig. 6.2b). This will be

achieved by recurrent evolutionary computation according to Fig. 6.5. embodied in the agents.

Table 6.1. Predictive rules for annual dynamics of *Microcystis* and *Oscillatoria* evolved from the Lake Kasumigaura database (Bobbin and Recknagel 2001)

Evolved Rules for Microcystis	Evolved Rules for Oscillatoria
IF (TEMP > 29 °C) AND	IF (NH4 < 236 µg/l) AND
IF (DTP > 74.2 µg/l) AND	IF (8.01 < pH < 9.37) AND
IF (pH > 8.15)	IF (60 < SECCHI < 103 cm) AND
THEN MICROCYSTIS >> 50,000 cells/ml	AND IF (DTP > 22.3 µg/l)
ELSE MICROCYSTIS < 50,000 cells/ml	THEN OSCILLATORIA >> 50,000 cells/ml
	ELSE OSCILLATORIA < 50,000 cells/ml

6.4.1.2 Adaptive Agents Bank

Natural ecosystems are characterised by redundancy in their composition and structure. They gain a certain degree of resilience to changing environmental conditions depending on the extent of redundancy. In order to develop adaptive agent models that gain such resilience to environmental changes, they need to have redundancy in their composition as well. Therefore, a bank of alternative and additional ecological agents for algal species will be developed occurring seasonally and locally in specific lakes under certain environmental conditions as reflected in the lake database (Tab. 6.3.). Evolutionary computation (EC) will be applied according to Fig. 6.5. to develop these algae-specific agents from the lake database that currently contains multivariate time-series of nine lakes different in eutrophication, climate and morphology. The range of conditions in the database will result in alternative agents for the same species/population resting in an agent bank.

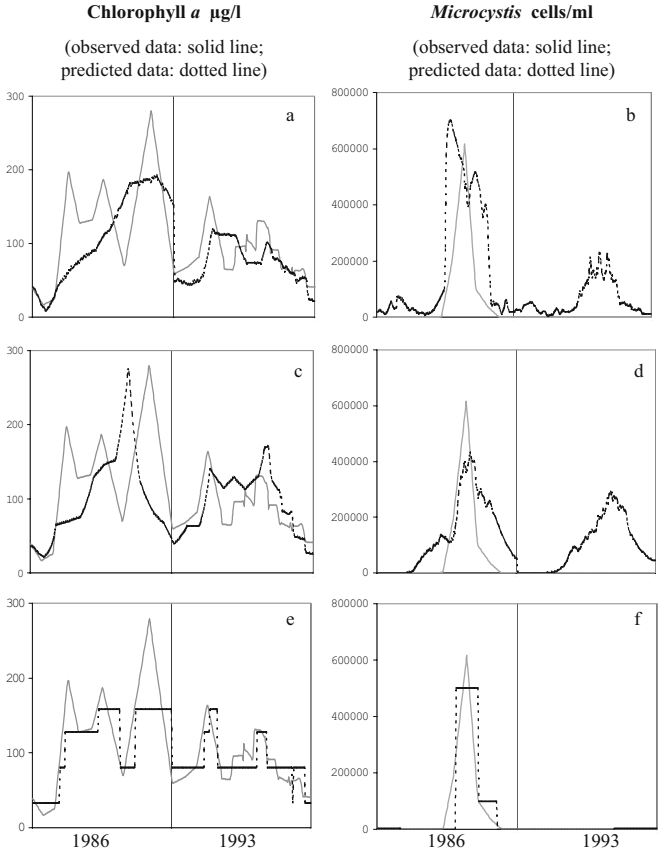


Fig. 6.4. 7-days-ahead prediction of Chlorophyll-a and *Microcystis* for Lake Kasumigaura (Japan) in 1986 and 1993 by ANN (a and b), EDE (c and d), and ER (e and f)

During simulations only those agents will be fired at a certain time that best suit occurring conditions but otherwise resting in the agent bank. Fired (emerging) agents simultaneously evolve based on EC in order to reach their optima (see Fig. 6.2b). This agents bank will be the key to enable the adaptive agents model to change the composition of agents during simulations by temporarily activating or resting agents (e.g. algal species) depending on excitatory or inhibitory environmental conditions.

Tab. 6.2. EDE and ER applied for 7-day-ahead predictions of *Microcystis* in Fig. 6.4.

Evolved Differential Equations	dA/dt = A(t) * (PHOT – RESP) - A(t) * (COP + CLAD) * 0.0001 – A(t) * (e / 5) PHOT = [a / b * T) * (0.025 * L / (c + 0.025 * L)) * (P/ A(t) /(d / X + P / X + d / A(t) + P / A(t)))] X = 5.76 * A(t) ^ 0.41 RESP = [(0.057 / b * T) + 0.3 * PHOT] where A = *Microcystis* biomass µg/l, PHOT=photosynthesis, RESP=respiration, L=photosynthetic active light, P=PO_4-P phosphate, X=auxiliary term, COP=biomass of crustacea copepoda mg/l, CLAD=biomass of crustacea cladocera mg/l, T = water temperature, a to e = constant parameters evolved to optimal values or functions
Evolved Rules	IF P >= 126 AND P <= 81.7 µg/l THEN *Microcystis* = 500,000 cells/ml ELSE IF pH >= 9.72 THEN *Microcystis* = 500,000 cells/ml EXCEPT IF T <= 19.5 ºC AND T >= 5.67 ºC THEN *Microcystis* = 3,000 cells/ml ELSE IF N/P >= 47.2 AND N/P <= 55.2 THEN *Microcystis* = 0 cells/ml ELSE IF S >= 95.5 cm THEN *Microcystis* = 3.5 cells/ml EXCEPT IF T >= 5.67 ºC AND T <= 15.7 ºC THEN *Microcystis* = 0 cells/ml OR IF N >= 1,110 µg/l THEN *Microcystis* = 0 cells/ml IF P >= 15.6 µg/l AND P <= 116 µg/l THEN *Microcystis* = 100,000 cells/ml EXCEPT IF T >= 26.7 ºC THEN *Microcystis* = 500,000 cells/ml EXCEPT IF S <= 160cm then *Microcystis* = 100,000 cells/ml ELSE IF N >= 757 µg/l AND N <= 1,690 µg/l THEN *Microcystis* = 0 cells EXCEPT IF T >= 15.7 ºC AND T <= 26.7 ºC THEN *Microcystis* = 3,000 cells/ml EXCEPT IF P <= 15.6 µg/l THEN *Microcystis* = 0 cells/ml ELSE IF T >= 15.7 AND T <= 26.7 ºC THEN *Microcystis* = 100,000 cells/ml EXCEPT IF S <= 160 cm AND S >= 74.4 cm THEN *Microcystis* = 0 cells/ml ELSE IF T >= 5.67 ºC AND T <= 15.7 ºC THEN *Microcystis* = 3,000 cells/ml ELSE *Microcystis* = 100,000 cells/ml. where S = Secchi Depth, T = water temperature, N = NO_3-N Nitrate

Tab. 6.3. Multivariate time series database of 9 lakes

Lakes	Years	Sampling Frequency	No of Water Quality Parameters	No of Phytoplankton Species/Groups	No of Zooplankton Species/Groups
Biwa (Japan)	1984 - 91	weekly to monthly	10	21	-
Burrinjuck (Australia)	1976 - 97	weekly to monthly	8	8	6
Kasumigaura (Japan)	1984 - 93	fortnightly to monthly	10	10	3
Myponga (Australia)	1970 - 97	weekly to monthly	10	25	-
Saidenbach (Germany)	1979 - 84	fortnightly	9	5	1
Soyang (Korea)	1984 - 99	monthly	7	12	12
Tuusulanjaervi (Finland)	1972 - 87	fortnightly to monthly	7	10	-
Veluvemeer (Holland)	1976 - 99	weekly to monthly	11	14	6
Wolderwijd (Holland)	1976 - 99	weekly to monthly	11	14	6

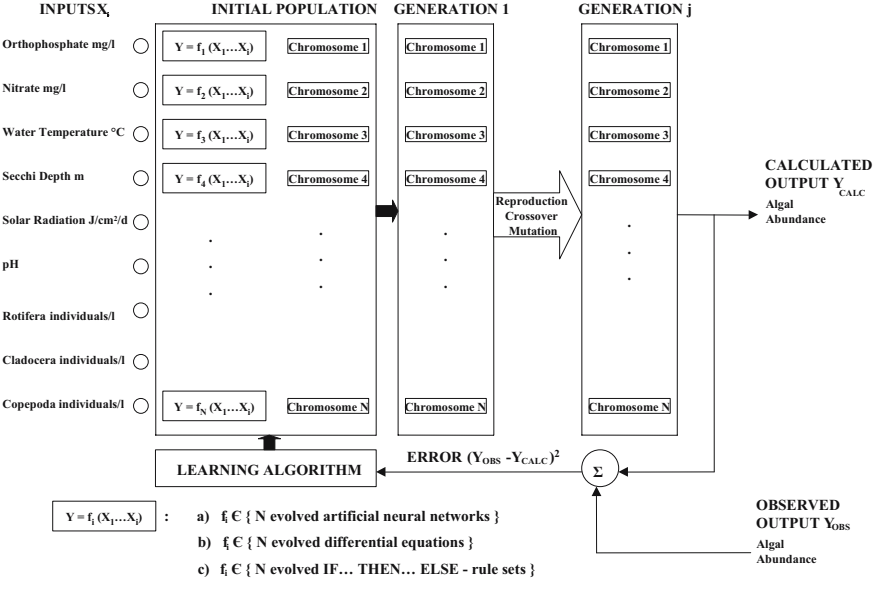


Fig. 6.5. Application of evolutionary computation to evolve ANN (a), EDE (b) and ER (c) from lake databases to be embodied in algal specific agents

6.4.2
Pelagic Food Web Simulation by Adaptive Agents

Adaptive agents simulation of pelagic food webs will be implemented according to Figure 6.6. Seven state variable-based agents will be considered initially to represent the following ecological entities: blue-green algae, green algae and diatoms, herbivorous and carnivorous zooplankton, planktivorous and piscivorous fish (see Figure 6.6a).

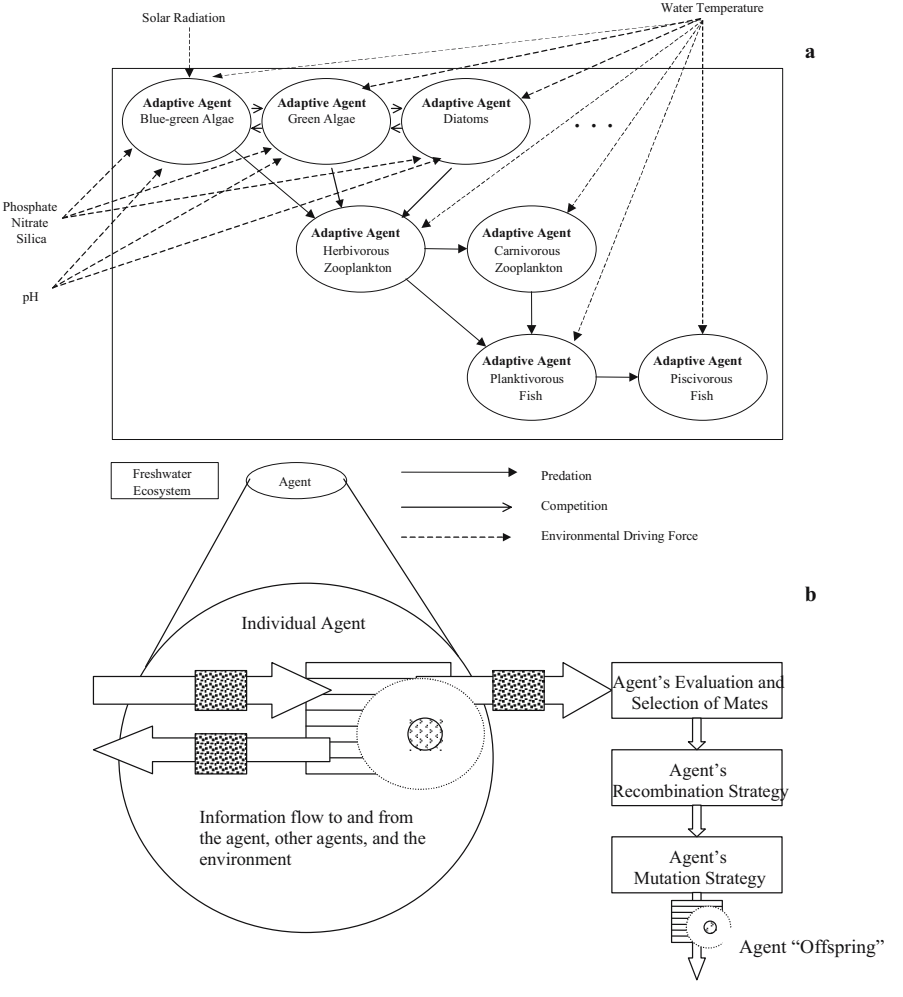


Fig. 6.6. Adaptive agents simulation of pelagic food-web dynamics

These seven agents interact by predation and competition, and are determined by environmental driving forces such as solar radiation, water temperature,

nutrient loadings. Each single agent is determined by EDE in order to maximise (adapt) their performance (abundance) in relation to current environmental conditions (nutrient loadings, light, temperature and abundance of competitors, predators or preys). EDE utilise evolutionary algorithms in order to steadily optimise parameter values and functions of the state variable-based agents by means of differential equations as used by Park et al. (1974) and Recknagel and Benndorf (1982). As a result, each agent adapts simultaneously to current environmental conditions by producing "offspring" agents based on its evaluation and selection of mates, recombination strategy and mutation strategy (see Figure 6.6b). Successful case studies on EDE have been conducted by Whigham and Recknagel (2001a, b) and Recknagel et al. (2002).

6.5 Conclusions

1. Adaptive agents (AA) provide a realistic framework for ecosystem simulation, evolving ecosystem structures and behaviours by emerging, submerging, interacting and evolving ecological entities.
2. Individual-based AA prove applicable to a spatially explicit simulation of highly simplified terrestrial food webs.
3. State variable-based AA where evolutionary computation is embodied appear to be relevant for simulations of aquatic food webs dynamics and plankton species interactions.
4. Embodiment of evolutionary computation in adaptive agents for aquatic species or functional groups can be achieved by evolving predictive rules (ER), differential equations (EDE) or artificial neural networks (ANN) from a diverse lake database.
5. Ecosystem simulation by state variable-based adaptive agents gains resilience to environmental change from an agent bank providing alternative agents for same species or functional groups evolved from a diverse lake database.
6. The presented concepts are currently tested by means of a multivariate time-series database for nine lakes different in climate, eutrophication and morphology.

Acknowledgements

I am very grateful to Michio Kumagai, Lake Biwa Research Institute, Japan, Myriam Bormans, CSIRO Land and Water, Australia, Noriko Takamura, National Institute for Environmental Studies, Japan, Mike Burch, Australian Water Quality Centre, Australia, Wolfgang Horn, Dresden University of Technology, Germany, Bomchul Kim, Kangwon National University, South Korea, Olli Varis, Helsinki University of Technology, Finland, and Diederik van der Molen, Dutch Institute

for Inland Water management, The Netherlands, for making invaluable data available from lakes in the order of Tab. 2. I also want to thank Bernard Patten, University of Georgia, U.S.A. for his critical but encouraging comments on the first draft of the paper.
This research was supported by the Australian research Council under the grant DP0345279.

References

Bobbin J, Recknagel F (2001) Knowledge Discovery for Prediction and Explanation of Blue-Green Algal Dynamics in Lakes by Evolutionary Algorithms. Ecol. Modelling 146, 1-3, 253-264

Booth G (1997) Gecko: a continuos 2-D world for ecological modeling. Artif. Life 3, 147-163

Forbes SA (1887) The lake as a microcosm. Bull.Sci.Ass., Peoria. Illinois, 77-87

Holland JH (1992) Adaptation in Natural and Artificial Systems. Addison-Wesley, New York

Holland JH (1998) Emergence. From Chaos to Order. Oxford University Press, Oxford, New York, Tokyo

Holland JH, Miller JH (1991) Artificial adaptive agents in economic theory. American Economic Review 81, 2, 365-370

Huston M, DeAngelis D, Post W (1988) New computer models unify ecological theory. BioScience 38, 10, 682-691

Jorgensen SE, Mejer H (1979) A holistic approach to ecological modelling. Ecol. Modelling 3, 39-61

Jorgensen SE (1986) Structural dynamics model. Ecol. Modelling 31, 1-9

Kaluzny S, Swartzman G (1985) Simulation experiments comparing alternative process formulations using factorial design. Ecol.Modelling 28, 181-200

Kreft JU, Booth G, Wimpeny JWT (1998) BacSim, a simulator for individual-based modelling of bacterial colony growth. Microbiology 144, 3275-3287

Lin F, Pai Y (2000) Using multi-agent simulation and learning to design new business processes. IEEE Transactions on Systems, Man, and Cybernetics 30, 3, 380-384

Park RA, O'Neill RV, Bloomfield JA, Shugart HH, Booth RS, Goldstein RA, Mankin JB, Koonce JF, Scavia D, Adams MS, Clesceri LS, Colon EM, Dettman EH, Hoopes JA, Huff DD, Katz S, Kitchell JF, Koberger RC, La Row EJ, McNaught DC, Petersohn L, Titus JE, Weiler PR, Wilkinson JW, Zahorcak CS (1974) A generalized model for simulating lake ecosystems. Simulation 33-50

Radtke E, Straskraba M (1980) Self-optimization in a phytoplankton model. Ecol. Modelling 9, 247-268

Railsback StF (2001) Concepts from complex adaptive systems as a framework for individual-based modeling. Ecological Modelling 139, 47-62

Recknagel F, Bobbin J, Whigham P, Wilson H (2002) Comparative application of artificial neural networks and genetic algorithms for multivariate time series modelling of algal blooms in freshwater lakes. Journal of Hydroinformatics 4, 2, 125-134

Recknagel F (1997) ANNA - Artificial Neural Network model predicting species abundance and succession of blue-green Algae. Hydrobiologia, 349, 47-57

Recknagel F, French M, Harkonen P, Yabunaka K (1997) Artificial neural network approach for modelling and prediction of algal blooms. Ecol. Modelling 96, 1-3, 11-28

Recknagel F, Benndorf J (1982) Validation of the ecological simulation model SALMO. Int. Revue ges. Hydrobiol. 67, 1, 113-125

Rigler FH, Peters RH (1995) Science and Limnology. Ecology Institute. Oldendorf

Schmitz OJ, Booth G (1997) Modelling food web complexity: The consequences of individual-based, spatially explicit behavioral ecology on trophic interactions. Evolutionary Ecology 11, 379-398

Straskraba M (1979) Natural control mechanisms in models of aquatic ecosystems. Ecol. Modelling 6, 305-322

Straskraba M, Gnauck A (1985) Freshwater Ecosystems, Modelling and Simulation. Elsevier, Amsterdam

Takamura N, Otsuki A, Aizaki M, Nojiri Y (1992) Phytoplankton species shift accompanied by transition from nitrogen dependence to phosphorus dependence of primary production in Lake Kasumigaura, Japan. Arch. Hydrobiol. 124, 129-148

Wan HA, Hunter A (1997) On artificial adaptive agents models of stock markets. Simulation 68, 5, 279-289

Whigham P, Recknagel F (2001) An Inductive Approach to Ecological Time Series Modelling by Evolutionary Computation. Ecol. Modelling 146, 1-3, 275-287

Whigham P, Recknagel F (2001) Predicting Chlorophyll-a in Freshwater Lakes by Hybridising Process-Based Models and Genetic Algorithms. Ecol. Modelling 146, 1-3, 243-251

Yao X Liu Y (1997) A new evolutionary system for evolving artificial neural networks. IEEE Transactions on Neural Networks, 8, 3, 694-713

Chapter 7

Bio-Inspired Design of Computer Hardware by Self-Replicating Cellular Automata

G. Tempesti · D. Mange · A. Stauffer · E. Petraglio

7.1 Introduction

The use of populations of artificial organisms has established itself as a useful tool for modelling natural systems. When this concept is applied to the optimization of computing systems, approaches such as evolutionary algorithms have also proven very useful in the search for good solutions to very complex problems. The research we have been conducting in the past years has explored a different use of the concept of artificial organisms in the very specific framework of the design of complex computer hardware.

The analogy between biology and electronics is not as farfetched as it might appear at a first glance. Aside from the more immediate parallel between the human brain and the computer, which has led to the development of fields such as artificial intelligence or neural networks, a certain degree of similarity exists between the genome (the hereditary information of an organism) and a computer program.

The genome consists of a one-dimensional string of data encoded in a base-4 system. The DNA (Deoxyribonucleic Acid), the macromolecule in which the genome is encoded, is a sequence of four bases: A (Adenine), C (Cytosine), G (Guanine), and T (Thymine). The information stored on the DNA is chemically decoded and interpreted to determine the function of a cell. A computer program is a one-dimensional string of data encoded in a base-2 system (0 and 1). Stored in an electronic memory circuit, it is interpreted to determine the function of a processor. Of course, carbon-based biology and silicon-based computing are different enough that no straightforward one-to-one relationship between the genome and a computer program (and indeed between any biological and computing process) can be established, except at a very superficial level. However, through careful interpretation, some basic biological concepts can be adapted to the design of computer systems, and some biological processes are indeed extremely interesting from a computer designer's point of view: for example, an organism's robustness (achieved by its healing processes) is unequaled in electronic circuits, while the natural process of evolution has produced organisms of a complexity which far exceeds that of modern computer systems.

The goal of our project, called *Embryonics* (Mange et al. 2000) for *embryonic electronics*, is then to try to exploit some of the remarkable features of natural organisms and of populations of such organisms in computer hardware. To this end, we have been studying how the concept of *self-replication* (historically known also as *self-reproduction*) both at the cellular and at the organism level can be applied to computer hardware.

Achieving self-replication in computer hardware, however, is a very complex technological challenge. Historically, this process has been studied though a layer of abstraction by exploiting the *cellular automata* (CA) model (Codd 1968; Wolfram 1994), a computational model which, while both ill-suited for hardware implementations and difficult to manipulate, nevertheless provides a relatively strict mathematical framework for the development of self-replicating structures. Within this model, the study of self-replication has a long, if not very eventful, history. After a very short introduction to the CA model, we will introduce the most salient points of this history before introducing the results of our own research.

7.2 Cellular Automata

Cellular automata are arrays of *elements*, or *cells*, whose behavior depends on the elements' state (while there is no theoretical limit to the number of dimensions of a cellular automaton, the implementations described herein are all two-dimensional). At regular, discrete intervals (*iterations*), the state of all elements is updated, depending on the current state of the element itself and that of its neighbors, according to a set of *transition rules*.

It should be noted that, to avoid conflicts with biological definitions, we shall not use the conventional term "cell" to indicate the parts of a cellular automaton, opting rather for the term "element" or, in the context of our project, "molecule". In fact, in biological terms, a *cell* can be defined as the smallest part of a living being which carries the complete blueprint of the being, that is the being's *genome*, a definition which is not met by the elements of a CA.

In order to illustrate the operation of cellular automata, we can examine one of the best-known (and simplest) two-dimensional CAs, commonly referred to as *Life* (Gardner 1970), an automaton where each element can be in one of only two states (alive or dead). The next state of an element depends on its current state and that of its eight closest neighbors (to the north, south, east, west, northeast, southeast, southwest, and northwest), and is calculated from a set of simple rules: if fewer than two elements in the neighborhood are alive, the next state is dead (death by starvation); if more than three elements in the neighborhood are alive, the next state is dead (death by overcrowding); if exactly three elements in the neighborhood are alive, then the next state is alive (birth); otherwise (i.e., if exactly two of the elements in the neighborhood are alive) the next state is equal to the current state (survival).

This very simple automaton, even if theoretically inspired by the behavior of populations of individuals, is obviously not very powerful: for example, the majority of initial configurations (the set of the states of all elements at iteration 0) lead either to an empty space or to a collection of small, isolated cyclic patterns. Complex and stable structures (Figure 7.1) can be created by putting together simple building blocks, but no methodology exists for their design.

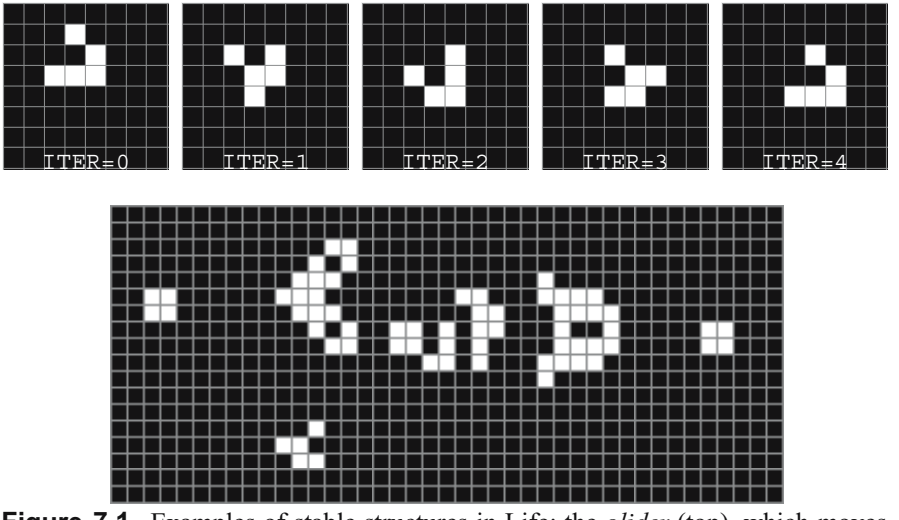


Figure 7.1. Examples of stable structures in Life: the *glider* (top), which moves diagonally through space, and the *glider gun* (bottom), which generates gliders at regular intervals.

To resume, a cellular automaton is defined by the following parameters:

- A number of *dimensions*, usually one or two, rarely three, and almost never four or more. All the automata used to model self-replication are two-dimensional, as is Life.
- A set of *states* (two in the case of Life) and an *initial configuration*, defining the state of all the elements of the array at iteration 0. While there is no theoretical limit to the number of states in an automaton, for practical considerations very few automata use more than a handful.
- A *neighborhood*, which specifies which neighbors will have an effect on an element's next state. By far the most common for two-dimensional automata are the neighborhood of 5 (the element itself plus its cardinal neighbors to the north, south, east, and west) and that of 9 (the element itself plus its neighbors to the north, south, east, west, northeast, southeast, southwest, and northwest). Life uses a neighborhood of 9.
- A collection of *transition rules*, used to compute an element's next state

depending on the neighborhood. The rules can be expressed either as an algorithm (as for Life above) or exhaustively as a lookup table. In the latter

case, the total number of rules necessarily to exhaustively define a cellular automaton is S^N, where S is the number of states and N the neighborhood (thus, to completely define Life, a lookup table of 2^9=512 rules would be required). In practice, the lookup table for a complex automaton (i.e., one with many states) can reach a very important size.

It should be clear that cellular automata are not a model which can easily be applied to digital hardware: the need for each element to access the transition rules, coupled with the large number of elements required for complex behavior, is a serious drawback for an electronic implementation.

These and other considerations have led researchers to tackle the issue of implementing populations of complex structures in hardware through a further level of abstraction, by defining a self-replicating mechanism that can, in principle, be applied to structures of arbitrary complexity.

7.3 Von Neumann's Universal Constructor

The field of self-replicating hardware was pioneered by John von Neumann (Asprey 1992). A gifted mathematician and one of the leading figures in the development of the field of computer engineering, von Neumann dedicated the final years of his life on what he called the theory of automata (von Neumann 1966). His research, which was unfortunately interrupted by von Neumann's untimely death in 1957, was inspired by the parallel between artificial automata, of which the paramount example are computers, and natural automata such as the nervous system, evolving organisms, etc.

In particular, Von Neumann, confronted with the lack of reliability of computing systems, turned to nature to find inspiration in the design of fault-tolerant computing machines. Natural systems are among the most reliable complex systems known to man, and their reliability is a consequence not of any particular robustness of the individual cells (or organisms), but rather of their extreme redundancy. The basic natural mechanism which provides such reliability is self-reproduction (the term used by von Neumann), both at the cellular level (where the survival of a single organism is concerned) and at the organism level (where the survival of the species is concerned).

In von Neumann's work, self-reproduction is always presented as a special case of universal construction, that is, the capability of building any machine given its description (Figure 7.2). This approach was maintained in the design of his cellular automaton, which it therefore much more than "just" a self-replicating machine. The complexity of its purpose is reflected in the complexity of its structure, based on three separate components:

- A memory tape, containing the description of the machine to be built, in the form of a one-dimensional string of elements. In the special case of self-reproduction, the memory contains a description of the universal constructor

itself (Figure 7.3).

- The constructor itself, a very complex machine capable of reading the memory tape and interpreting its contents.
- A constructing arm, directed by the constructor, used to build the offspring (i.e., the machine described in the memory tape). The arm moves across the space and sets the state of the elements of the offspring to the appropriate value.

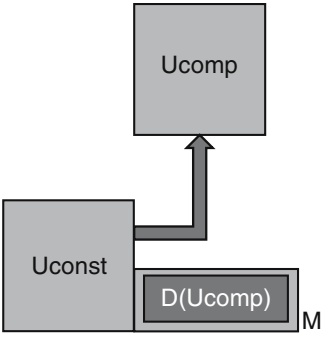


Figure 7.2. Von Neumann's universal constructor *Uconst* can build a specimen of any machine (e.g., a universal Turing machine *Ucomp*) given its description *D(Ucomp)*.

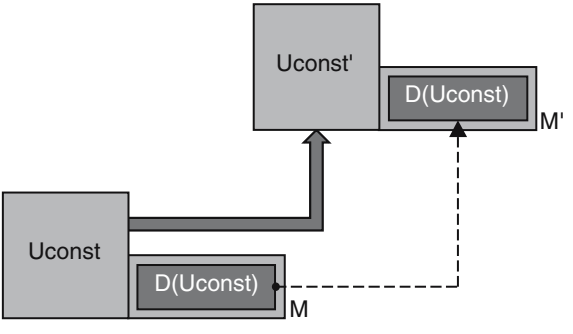


Figure 7.3. Given its own description *D(Uconst)*, von Neumann's universal constructor is capable of self-replication.

The implementation as a cellular automaton is no less complex. Each element has 29 possible states, and thus, since the next state of an element depends on its current state and that of its four cardinal neighbors, 29^5=20,511,149 transition rules are required to exhaustively define its behavior. If we consider that the size of von Neumann's constructor is of the order of hundreds of thousands elements, we can easily understand why a hardware realization of such a machine is not really feasible.

We should mention that von Neumann went one step further in the design of his universal constructor. If we consider the universal constructor from a biological viewpoint, we can associate the memory tape with the genome, and thus the entire constructor with a single cell (which would imply a parallel between the automaton's elements and molecules). However, the constructor, as we have described it so far, has no functionality outside of self-reproduction. Von Neumann recognized that a self-replicating machine would require some sort of functionality to be interesting from an engineering point of view, and postulated the presence of a universal computer (in practice, a universal Turing machine, an automaton capable of performing any computation) alongside the universal constructor (Figure 7.4). Von Neumann's constructor can thus be regarded as a *unicellular organism*, containing a genome stored in the form of a memory tape, read (transcription) and interpreted (translation) by the universal constructor both to determine its operation and to direct the construction of a complete copy of itself.

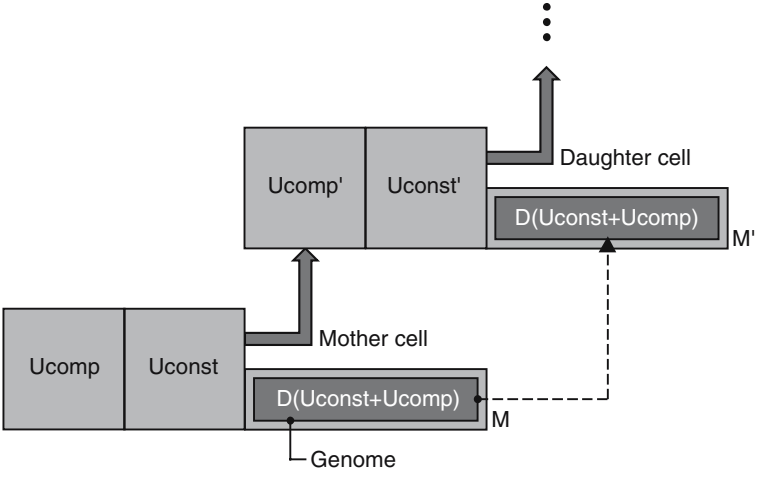


Figure 7.4. By extension, Von Neumann's universal constructor can include a universal computer and still be capable of self-replication.

As we mentioned, the dimensions of von Neumann's universal constructor are substantial; it has thus never been physically implemented and has been simulated only partially (Pesavento 1995). To the best of our knowledge, the only attempt to implement a complete specimen is the current work of Buckley (Buckley 2004) whose latest result specifies that the universal constructor (without its tape) is bounded by a region of 751x1048 = 787,048 cells.

The impossibility of achieving a physical realization did not however deter some researchers from trying to continue and improve von Neumann's work. Arthur Burks, for example, in addition to editing von Neumann's work on self-replication (Burks 1970; von Neumann 1966), also made several corrections and advances in the implementation of the cellular model. Codd (1968), by altering the

states and the transition rules, managed to simplify the constructor by a considerable degree. However, without in any way lessening these contributions, we can say that no major theoretical advance in the research on self-reproducing automata occurred until Langton (1984) opened a second stage in this field of research.

7.4 Self-replicating Loops

While the early history of the theory of self-replicating machines is basically the history of John von Neumann's thinking on the matter, a practical implementation requires a sharply different approach. In order to construct a self-replicating automaton simpler than this of von Neumann, Langton (Langton 1984) adopted more liberal criteria. He dropped the condition that the self-replicating unit must be capable of universal construction and computation.

Langton proposes a configuration in the form of a loop (Figure 7.5), endowed notably of a constructing arm and of a replication program or genome, which turns counterclockwise. After 151 time steps, the original loop (mother loop) produces a daughter loop, thus obtaining the self-replication of Langton's loop.

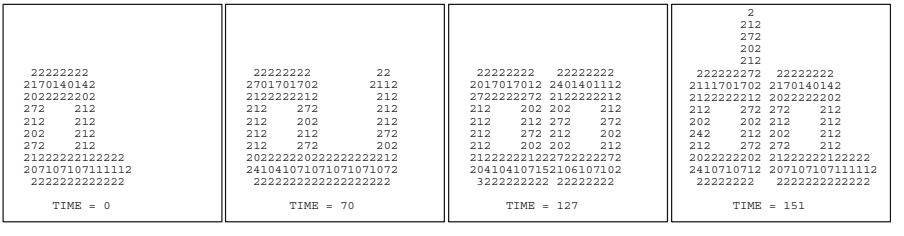


Figure 7.5. Langton proposes a self-replicating machine in the form of a loop.

According to the biological definition of a cell as the smallest part of a living being which carries the complete genome, we end up with the following observations.

- Langton's self-replicating loop is a unicellular organism: its genome requires 28 molecules and is a subset of the complete loop which requires 94 molecules.
- The size of Langton's loop is perfectly reasonable, since it requires 94 molecules, thus allowing complete simulation.
- There is no universal construction nor calculation: the loop does nothing but replicate itself. Langton's self-replicating loop represents therefore a special case of von Neumann's self-replication of a universal constructor. The loop is a non-universal constructor, capable of building, on the basis of its genome, a single type of machine: itself.

As did von Neumann, Langton emphasized the two different modes in which information is used, interpreted (*translation*) and uninterpreted (*transcription*). In

his loop, translation is accomplished when the instruction signals are executed as they reach the end of the construction arm, and upon collision of signals with other signals. Transcription is accomplished by duplication of signals at the arm junctions.

More recently, Byl (1989) proposed a simplified version of Langton's automaton. Last but not least Reggia *et al.* (1993) discovered that having a sheath surrounding the data paths of the genome was not essential, and that its removal led to smaller self-replicating structures which also have simpler transitions functions. Moreover, they found that relaxing the strong symmetry requirement consistently led to transition functions that required fewer rules than the corresponding strong symmetry version.

All the previous loops lack any computing and constructing capabilities, their sole functionality being that of self-replication. Lately, new attempts have been made to redesign Langton's loop in order to embed such calculation possibilities. Tempesti's loop (Tempesti 1995) is thus a self-replicating automaton, with an attached executable program that is duplicated and executed in each of the copies. This was demonstrated for a simple program that writes out (after the loop's replication) "LSL", acronym of the Logic Systems Laboratory. Finally, Perrier *et al.*'s self-replicating loop (Perrier *et al*. 1996) shows some kind of universal computational capabilities. The system consists of three parts, loop, program, and data, all of which are replicated, followed by the program's execution on the given data.

These improvements notwithstanding, CAs remain a model which is ill-adapted, at least in its conventional form, to the design of computer hardware. In our project, we have developed an hybrid approach which couples CA with some of the most interesting features of today's programmable circuits, generally known as FPGAs (Sanchez 1996; Trimberger 1994), to efficiently implement self-replication in hardware systems.

7.5 Self-replication in the Embryonics Project

7.5.1 Embryonics

The main goal of the Embryonics (embryonic electronics) project (Mange *et al*. 2000) is to realize, in an integrated circuit, systems embedding features inspired by the behavior of multi-cellular biological organisms. These bio-inspired systems would then possess some of the characteristics of organisms, such as the capability to self-repair (heal) and to self-replicate to create populations of complex entities. To maintain the analogy to living systems, our approach is based on four hierarchical levels of organization (Figure 7.6). Within this hierarchy, we shall call

organism a computing system dedicated to the execution of an arbitrarily complex application. An organism is then composed of identical parts called *cells*, where each cell is a simple, application-specific processor. Each cell is itself composed of a finite number of elements referred to as *molecules*.

At the core of the system is the cell: an artificial organism is realized by a matrix of identical cells distributed over the nodes of a regular two-dimensional grid. Each cell contains a small processor and a memory in which the genome program (identical for all the cells) is stored. In this multicellular organization only the state of a cell (i.e. the contents of its registers) can differentiate it from its neighbors.

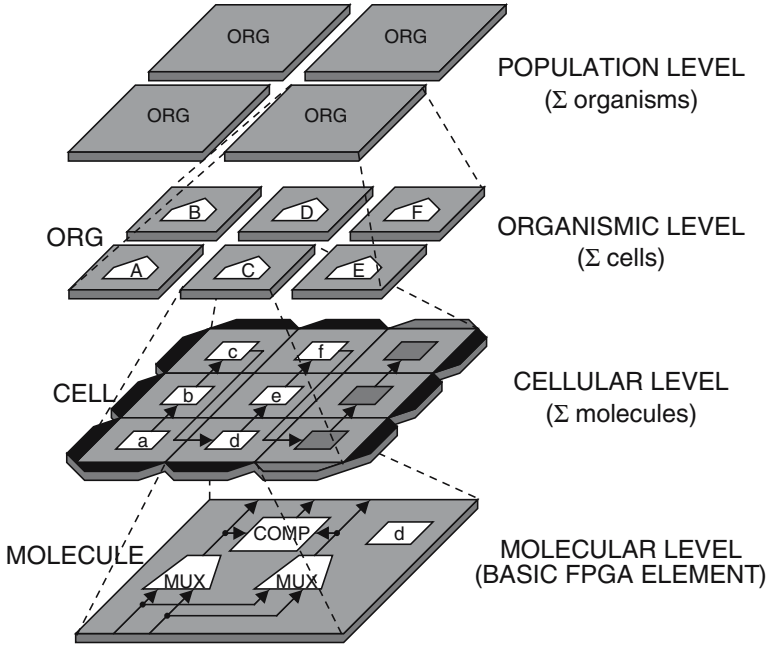


Figure 7.6. The four hierarchical levels of organization in Embryonics.

In the organism each cell realizes a unique function, defined by a sub-program called the *gene*, which is a part of the genome. Each cell knows its position (i.e. X and Y coordinates) in the organism and uses it to define which gene of the genome it has to execute. In Figure 7.7 the genes are labelled A to F for coordinates $(X,Y)=(1,1)$ to $(X,Y)=(3,2)$ (note that, in the figure, only the expressed gene is represented in the cell).

In this context, an artificial organism is capable of replicating itself if there is enough free space in the silicon circuit (at least six cells in the example of Figure 7.7) to contain the new daughter organism and if the calculation of the

coordinates produces a cycle (X=1→2→3→1... and Y=1→2→1... , implying X=(WX+1)*modulo*3 and Y=(SY+1)*modulo*2). Since each cell stores the same information (i.e. the genome program), the cycling of the coordinates causes the repetition of the same pattern of genes: therefore, in a sufficiently large array of cells, the self-replication process can be repeated for any number of specimens in the X and/or the Y axes.

This self-replication of the organism, achieved through the cycling of the cell's coordinates, is then an immediate consequence of the self-replication of the artificial cells. In fact, a cell has to self-replicate to obtain a collection of identical cells, which will compose the first artificial organism. The crucial hardware mechanism necessary to obtain populations of organisms is therefore the same as the one necessary to obtain a single multi-cellular organism.

The self-replication of an arbitrarily complex cell is a task which, at first sight, bears a close resemblance to the self-replication of structures such as von Neumann's universal constructor and Langton's loop. When the problem is observed in detail, however, the requirements of an efficient hardware implementation introduce several important differences.

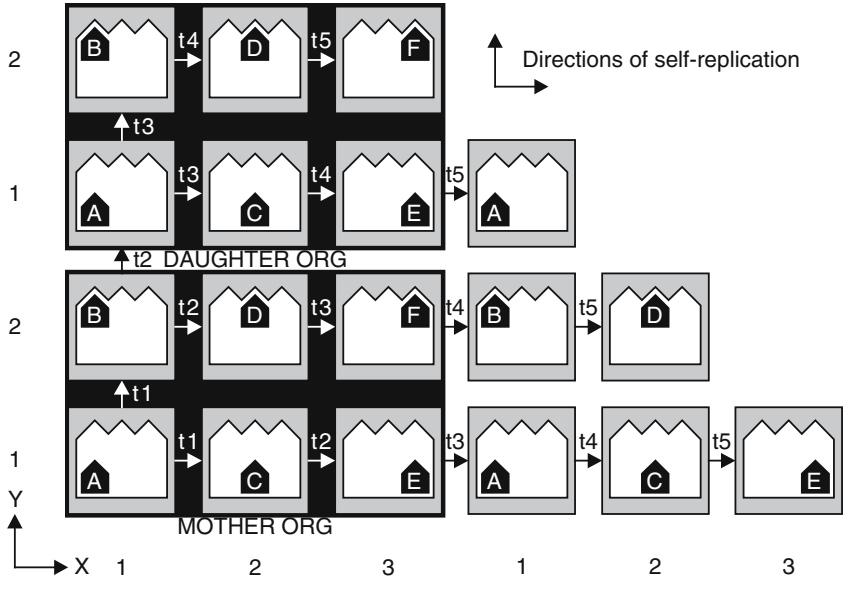


Figure 7.7. Self-replication of a 6-cell organism in a limited homogeneous array of 6x4 cells (situation at the time $t5$ after 5 cellular divisions).

As mentioned above, a cell is not the hardware primitive of our systems, but is divided into a number of small identical parts called molecules. These molecules correspond to the basic elements of a programmable logic circuit (commonly referred to as a *field programmable gate array* or FPGA). FPGAs (Figure 7.8) are

circuits composed by a two-dimensional grid of simple logic elements that can be *configured* to perform one of several different functions and to connect to each other in order to implement arbitrarily complex circuits. They are usually reconfigurable (i.e., their configuration can be erased and replaced by another to implement a different kind of circuit), which makes them ideal to implement the kind of adaptive circuits necessary for bio-inspired circuits.

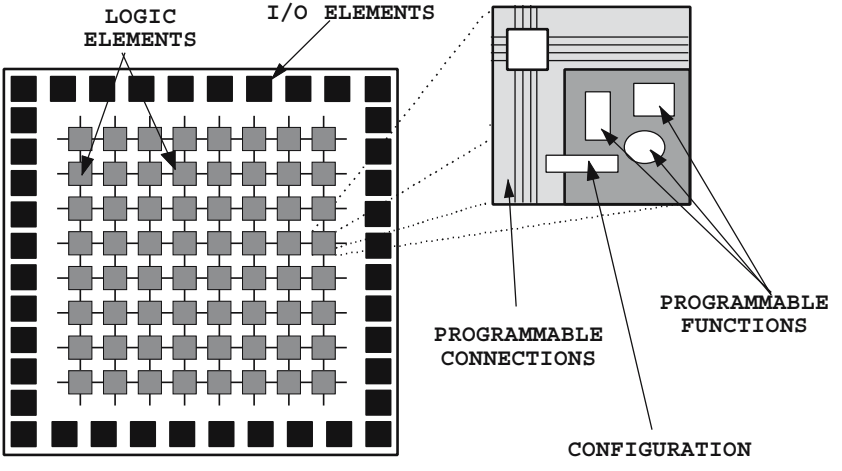


Figure 7.8. Basic structure of a generic FPGA circuit.

The configuration of the complete circuit is the sum of the configurations of the single elements: we are then faced with a two-dimensional array of identical elements whose function is determined by their state (the contents of the local configuration memory). This structure is very similar to a CA and indeed the configuration memories can be designed to behave as a CA. Nevertheless, some important differences exist.

Probably the most important difference lies in the fact that, unlike conventional CAs, the number of states of an FPGA element is extremely large: a configuration memory cannot in practice be smaller than 16 bits (2^{16}=65,536 states) and is often in the range of 64 to 128 bits. Obviously, the behavior of this kind of CA cannot be described in a lookup table and alternative methods of defining the state transitions must be found.

To address this issue, we developed both a novel approach to the design of hardware-friendly cellular automata, described in detail elsewhere (Stauffer and Sipper 2003), and a new CA-based algorithm that allows us to obtain self-replication of arbitrarily large machines in an FPGA.

7.5.2
The Tom Thumb Algorithm

Our algorithm, the *Tom Thumb algorithm* (Mange *et al.* 2004a; Mange *et al.* 2004b), aims at implementing in silicon, through self-replication, a process similar to cellular division in multi-cellular organisms. The algorithm exploits the use of FPGAs as the molecular substrate on which the cells are built (an element of the FPGA will then represent a molecule in this approach).

Before describing our new algorithm for the division of an artificial cell, let us remember the roles that cellular division plays in the existence of living organisms (Campbell, Reece and Mitchell 1999) (p. 206): "When a unicellular organism divides to form duplicate offspring, the division of a cell reproduces an entire organism. But cell division also enables multicellular organisms, including humans, to grow and develop from a single cell, the fertilized egg. Even after the organism is fully grown, cell division continues to function in renewal and repair, replacing cells that die from normal wear and tear or accidents. For example, dividing cells in your bone marrow continuously supply new blood cells. The reproduction of an ensemble as complex as a cell cannot occur by mere pinching in half; the cell is not like a soap bubble that simply enlarges and splits in two. Cell division involves the distribution of identical genetic material (DNA) to two daughter cells. What is most remarkable about cell division is the fidelity with which the DNA is passed along, without dilution, from one generation of cells to the next. A dividing cell duplicates its DNA, allocates the two copies to opposite ends of the cell, and only then splits into two daughter cells".

In conclusion, we can summarize the two key roles of cell division:

- The construction of two daughter cells in order to grow a new organism or to repair an already existing one (genome *translation*).
- The distribution of an identical set of chromosomes in order to create a copy of the genome from the mother cell aimed at programming the daughter cells (genome *transcription*).

Through an example of a minimal cell made up of four artificial molecules, we will show how the Tom Thumb algorithm can construct both the daughter cells and the associated genomes. A tissue of such molecules will in the end be able to constitute a multicellular organism and, with sufficient space, a population of such organisms.

7.5.2.1
Construction of the Minimal Cell

The minimal cell compatible with our algorithm is made up of four molecules, organized as a square of two rows by two columns (Figure 7.9.). Each molecule is able to store in its four memory positions four hexadecimal characters of our artificial genome, and the whole cell thus embeds 16 such characters. It should be

noted that our algorithm is perfectly scalable, i.e., it can be extended to larger molecules (more memory positions) and larger cells (more molecules) without any alteration to its behavior.

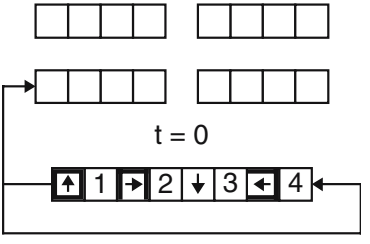
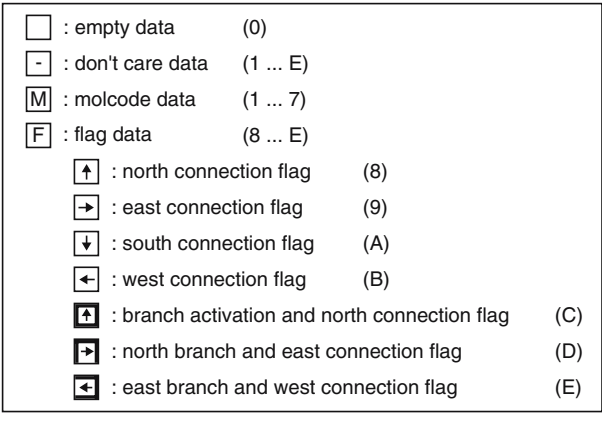
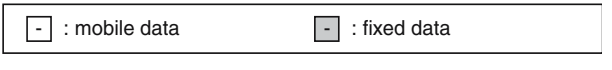


Figure 7.9. The minimal cell (2x2 molecules) with its genome at the start (t=0).

The original genome for the minimal cell is organized as a string of eight hexadecimal characters, i.e. half the number of characters in the cell, moving counterclockwise by one character at each time step (t=0,1,2,...). At startup, the original copy of the genome is stored in a register outside of the circuit, connected to the lower-left molecule in the array.


: empty data (0)
: don't care data (1 ... E)
: molcode data (1 ... 7)
: flag data (8 ... E)
: north connection flag (8)
: east connection flag (9)
: south connection flag (A)
: west connection flag (B)
: branch activation and north connection flag (C)
: north branch and east connection flag (D)
: east branch and west connection flag (E)

(a)



(b)

Figure 7.10. The 15 characters forming the alphabet of an artificial genome. (a) Graphical and hexadecimal representations of the 15 characters. (b) Graphical representation of the status of each character.

The 15 hexadecimal characters composing the alphabet of our artificial genome are detailed in Figure 7.10. They are either *empty data* (0), *molcode data*

(for molecule code data, from 1 to 7) or *flag data* (from 8 to E). Molcode data will be used for configuring our final artificial organism (they correspond to the configuration of the FPGA element in conventional circuits), while flag data are necessary for constructing the skeleton of the cell. Furthermore, each character is given a status and will eventually be *mobile data*, indefinitely moving around the cell, or *fixed data*, definitely trapped in a memory position of a molecule.

At each time step, a character of the original genome is shifted from right to left and simultaneously stored in the lower leftmost molecule (Figures 7.9. and 7.11.). Note that, due to our algorithm, the first, third, ... character of the genome (i.e. each odd character) is always a flag *F*, while the second, fourth, ... character (i.e. each even character) is always a molcode *M*. The construction of the cell, i.e. storing the fixed data and defining the paths for mobile data, depends on two major patterns (Figure 7.12.):

- If the two, three or four rightmost memory positions of a molecule are empty (blank squares), the characters are shifted by one position to the right (shift data).
- If the rightmost memory position is empty, the characters are shifted by one position to the right (load data). In this situation, the rightmost *F*' and *M*' characters are trapped in the molecule (fixed data), and a new connection is established from the second leftmost position toward the northern, eastern, southern or western molecule, depending on the fixed flag information (*F*' = 8 or C, 9 or D, A, B or E).

At time t=16, 16 characters, i.e. twice the contents of the original genome, have been stored in the 16 memory positions of the cell (Figure 7.11.). Eight characters are fixed data, forming the phenotype of the final cell, and the eight remaining ones are mobile data, composing a copy of the original genome, i.e. the genotype. Both *translation* (i.e. construction of the cell) and *transcription* (i.e. copy of the genetic information) have been therefore achieved.

The fixed data trapped in the rightmost memory position of each molecule remind us of the pebbles left by Tom Thumb to remember his way in the famous fable and gives its name to the algorithm.

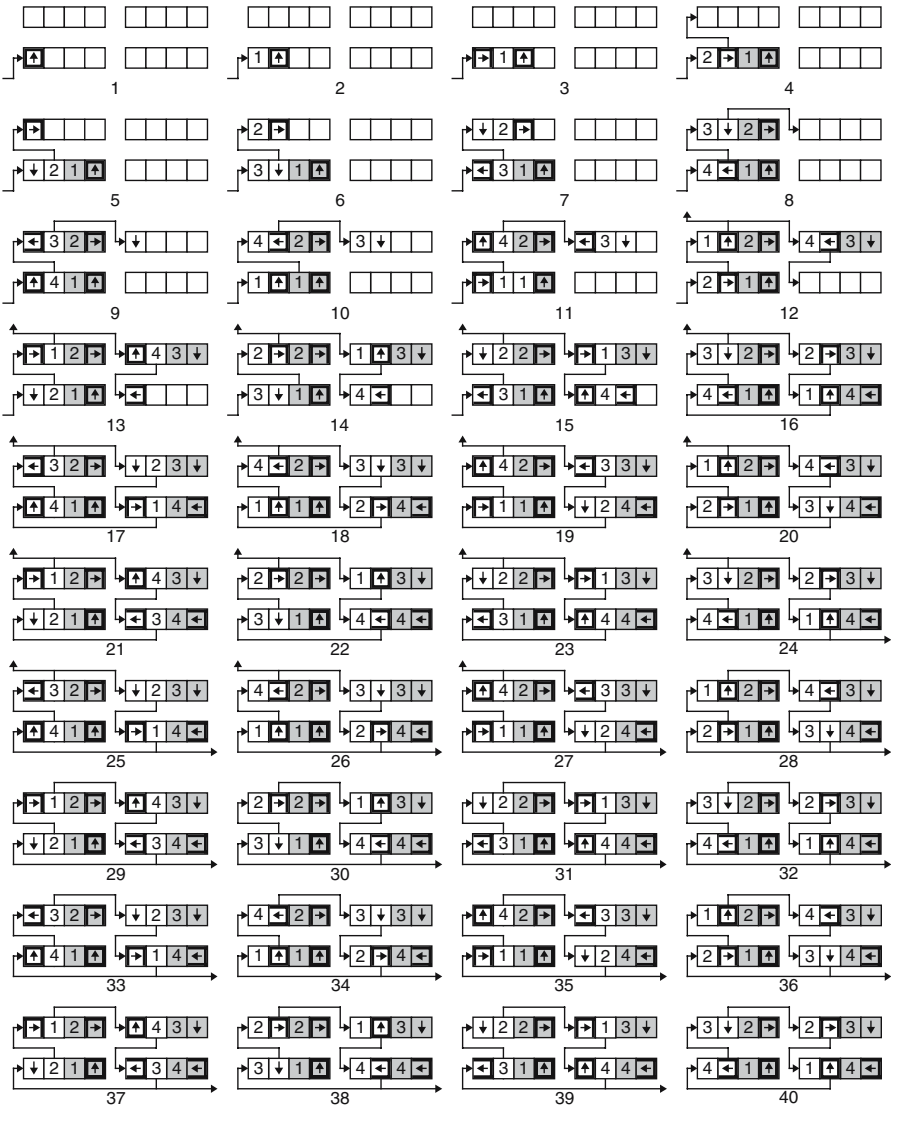


Figure 7.11. Constructing the minimal cell.

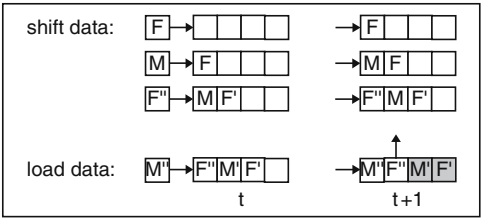


Figure 7.12. The two memory patterns for constructing a cell.

7.5.2.2 Growth and Self-replication

In order to grow an artificial organism in both horizontal and vertical directions, the mother cell should be able to trigger the construction of two daughter cells, northward and eastward.

At time t=11 (Figure 7.11.), we observe a pattern of characters which is able to start the construction of the northward daughter cell; the upper leftmost molecule is characterized by two specific flags, i.e. a fixed flag indicating a north branch (F = D) and the branch activation flag (F = C). This pattern is also visible in Figure 7.13. (northward signal, third row). The new path to the northward daughter cell will start from the second leftmost memory position.

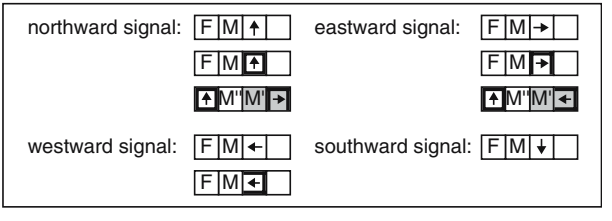


Figure 7.13. Patterns of characters triggering the paths to the north, east, south and west molecules.

At time t=23, another particular pattern of characters will start the construction of the eastward daughter cell; the lower rightmost molecule is characterized by two specific flags, i.e. a fixed flag indicating an east branch (F = E), and the branch activation flag (F = C). This pattern appears also in Figure 7.13. (eastward signal, third row). The new path to the eastward daughter cell will start from the second leftmost memory position.

The other patterns in Figure 7.13. are needed for constructing the inner paths of the minimal cell (Figure 7.11.) as well as cells more complex than the minimal cell.

Once the system is in place, the self-replication of the cell can occur indefinitely, as long as there is sufficient space in the circuit (Figure 7.14).

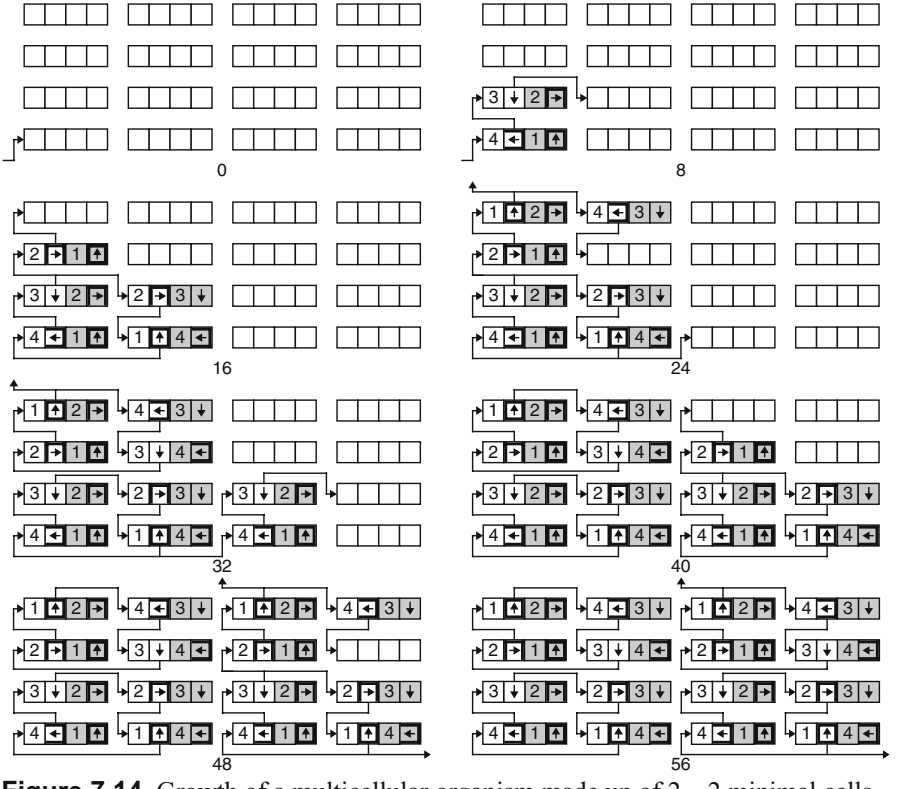
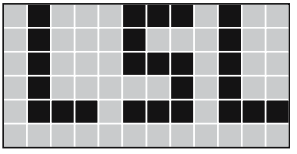


Figure 7.14. Growth of a multicellular organism made up of 2 2 minimal cells.

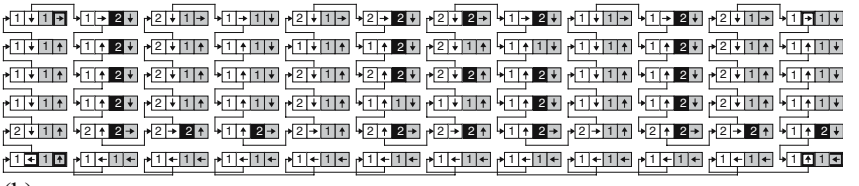
7.5.2.3
The LSL Acronym Design Example

In Tempesti (1995), we have already shown how to embed the acronym "LSL" (for Logic Systems Laboratory) into a self-replicating loop implemented on a classical cellular automaton. Thanks to a "cut-and-try" methodology and a powerful simulator, we were able to carry out the painful derivation of over ten thousand rules for the basic cell.

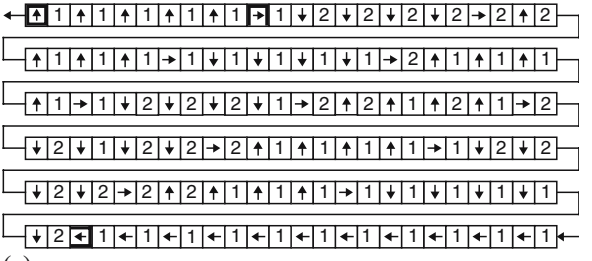
Unlike in that heuristic method, the same example can be designed in a straightforward and systematic way using the Tom Thumb algorithm.



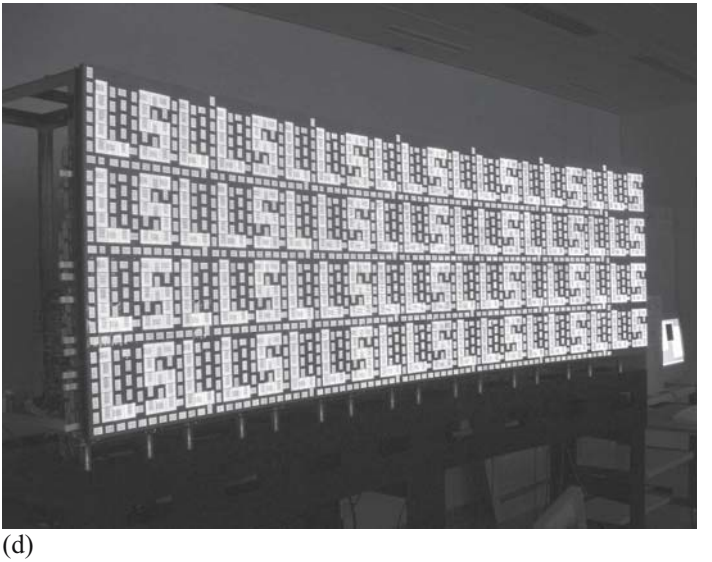
(a)



(b)



(c)



(d)

Figure 7.15. Self-replication of the "LSL" acronym. (a) Original specifications (LSL = Logic Systems Laboratory). (b) The 12x6=72 molecules of the basic cell. (c) Genome. (d) BioWall implementation displaying both the genotypic path and the phenotypic shape (Photograph by E. Petraglio).

The "LSL" acronym is first represented in a rectangular array of 12 columns by 6 rows (Figure 7.15 a). While the number of rows is indifferent, the number of columns should be even in order to properly close the loop (Figure 7.15 b). The cell is therefore made up of 12x6=72 molecules connected according to the pattern in Figure 7.15 b: bottom-up in the odd columns, top-down in the even columns, with the lower row reserved for closing the loop. It is then simple to define all the flags in the rightmost memory position of each molecule (grey characters in Figure 7.15 b).

Among the 72 molecules, 25 are used to display the three letters "L", "S" and "L", and are given the character "2" as molcode (black data in Figure 7.15 a and 7.15 b), while 47 are blank (molcode "1").

The detailed information of the final genome, i.e. 72x2=144 hexadecimal characters (Figure 7.15 c), is derived by reading clockwise the fixed characters (black and grey characters in Figure 7.15 b) of the whole loop, starting with the lower molecule of the first column.

Last, it was possible to embed our basic molecule in each of the 2000 field-programmable gate arrays of the BioWall (Tempesti *et al.* 2002) and to show the rather spectacular self-replication of our original cell (equivalent to a unicellular artificial organism), the "LSL" acronym, towards both vertical and horizontal directions (Figure 7.15 d), obtaining a population of identical organisms.

The LSL acronym design example can be easily generalized to produce the following algorithm:

1. Divide the given problem in a rectangular array of *C* columns by *R* rows. While the number of rows *R* is indifferent, the number of columns *C* should be even in order to properly close the loop.
2. Define all the flags in the rightmost memory position of each molecule according to the following patterns: bottom-up in the odd columns and top-down in the even columns, with the lower row reserved for closing the loop.
3. Complete the path by adding the branch activation and north connection flag (C) in the rightmost memory position of the lower molecule of the first column, the north branch and east connection flag (D) in the rightmost memory position of the upper molecule of the first column, and the east branch and west connection flag (E) in the rightmost memory position of the lower molecule of the last column, in order to trigger the two daughter loops northwards and eastwards respectively.
4. According to the original specifications, complete all the molcode data in the second rightmost memory position of each molecule. These molcode data constitute the phenotypic information of the artificial cell.
5. The detailed information of the final genome, i.e. the genotypic information of the artificial cell, is derived by reading clockwise along the original path the fixed characters of the whole loop, i.e. the two rightmost characters of each molecule, starting with the lower molecule of the first column. The genotypic information, or artificial genome, is used as the configuration string of the artificial cell and will eventually take place in the two leftmost memory positions of each molecule.

This methodology is made possible by modifying the CA paradigm into a more hardware-friendly format. We call this approach the *data and signals cellular automaton* (DSCA) (Stauffer and Sipper 2003). This kind of systems, which can

easily be transformed into conventional CAs (albeit ones that are much too complex to be handled conventionally), relies on a definition of the transition rules which is fundamentally different from the norm: rather than by referencing a lookup-table depending on the element's neighborhood, the transitions occur as a consequence of signals which depend on the elements' state. This approach allows us on one hand to handle elements with very large numbers of states (a necessary condition to translate CAs into the world of programmable circuits) and on the other hand to easily adapt the system when the specifications change (a crucial feature for the design of complex machines that is missing from conventional CAs).

7.5.2.4
Universal Construction

In his original contribution (von Neumann 1966), von Neumann defined construction (or constructibility) as the capability of constructing, i.e. assembling and building from appropriately defined "raw materials", an automaton using another automaton, the constructor. More precisely, the constructor, a two-dimensional automaton, is able to build in the two-dimensional array defined by von Neumann a specimen of another automaton described by a one-dimensional string of characters (the artificial genome) stored into the tape of the constructor.

According to von Neumann (1966), a constructor is endowed with universal construction if it is able to construct every other automaton, i.e. an automaton of any dimensions. This concept is pointed out by Freitas and Merkle (2004), where construction universality implies the ability to manufacture any of the finitely sized machines which can be formed from specific kinds of parts, given a finite number of different kinds of parts but an indefinitely large supply of parts of each kind.

If we assume (1) the existence of an array, as large as desired, of molecules and (2) the existence of a string of characters, as large as desired, the artificial genome, we claim that we are able to construct a computing machine of any dimensions into the array. Remember that the molcode data *M*, limited to the range 1 ... 7, may be directly used, as in the previous example, for displaying the given specifications or may configure any kind of field-programmable gate array aimed at defining a more complex digital architecture. There are only two restrictions involved by our actual implementation.

- The number of rows and/or columns should be even, in order to properly close the loop.
- For any artificial organism characterized by a molcode alphabet greater than 1 ... 7, we would be led to slightly modify the architecture of the molecule and use larger registers (with more than 4 bits).

If these two simple conditions are met, we can embed onto an array of molecules any array of boolean (octal, hexadecimal) values and observe the self-replication of the original pattern.

On the other hand, we have already shown that a universal Turing machine may be embedded in a regular array of identical cells (Restrepo and Mange 2001), themselves decomposed and implemented onto a regular array of molecules. Our new loop with universal construction can therefore verify *universal computation*, thus meeting the two basic properties of the historical self-replicating cellular automaton designed by von Neumann (1966), i.e. *universal construction and computation*.

7.6 Conclusion

In this work, we presented some solutions to the implementation of the biological concepts of organisms and populations in the context of computer hardware through CA-based mechanisms. Unlike more conventional approaches based on the use of cellular automata to *model* biological systems, the emphasis in these systems is on drawing *inspiration* from nature to design computing machines endowed with some of the remarkable properties of biological entities.

Several years before the publication of the historical paper by Watson and Crick (1935) revealing the existence and the detailed architecture of the DNA double helix, von Neumann was already able to point out that a self-replicating machine necessitated the existence of a one-dimensional description, the genome, and a universal constructor able to both interpret (translation process) and copy (transcription process) the genome in order to produce a valid daughter organism. Self-replication will allow not only to divide a mother cell (artificial or living) into two daughter cells, but also to grow and repair a complete organism. Self-replication is now considered as a central mechanism indispensable for those circuits which will be implemented through the nascent field of nanotechnologies (Drexler 1992; Roco and Bainbridge 2002).

From von Neumann's seminal work (von Neumann 1966) to Langton's loop (Langton 1984), cellular automata have been the framework of choice for the study of self-replicating machines in computer science. In our project, we analyzed the features of these systems in view of finally implementing self-replication in silicon. In particular, to step beyond the limitations imposed by the implicit complexity of cellular automata, we shaped our Embryonics project around artificial multicellular organisms based on the growth of a cluster of cells, themselves produced by cellular division (Macias and Durbeck 2002; Mange *et al.* 2000). The same mechanism is capable of producing populations of machines which could then be used in the context of evolution and to carry out massively parallel computation (Chou and Reggia 1998).

A major by-product of this research is the introduction of a new kind of cellular automaton, the data and signals cellular automaton (DSCA) (Stauffer and Sipper 2003), decomposed in a processing and a control units, which allows for a systematic and straightforward design methodology which has been sorely lacking.

We showed how this concept can be applied, though the Tom Thumb algorithm, to the self-replication of arbitrarily complex circuits.

Acknowledgments

This work was supported in part by the Swiss National Science Foundation under grant 20-100049.1 and by the Leenaards Foundation, Lausanne, Switzerland.

References

Asprey W (1992) John von Neumann and the Origins of Modern Computing. The MIT Press, Cambridge, MA

Buckley WR (2004) On the complete specification of a von Neumann 29-state self-replicating cellular automaton. Private communication, wrb@wrbuckley.com

Burks A, editor (1970) Essays on Cellular Automata. University of Illinois Press, Urbana

Byl J (1989) Self-reproduction in small cellular automata. Physica D, 34,295–299

Campbell NA, Reece JB, Mitchell, LG (1999) Biology, 5th edition. Benjamin/Cummings, Menlo Park, NJ

Chou HH, Reggia JA (1998) Problem solving during artificial selection of self-replicating loops. Physica D 115,3-4, 293–312

Codd EF (1968) Cellular Automata. Academic Press, New York

Drexler KE (1992) Nanosystems: Molecular Machinery, Manufacturing, and Computation. John Wiley, New York

Freitas RA, Merkle RC (2004) Kinematic Self-Replicating Machines. Landes Bioscience, Georgetown, TX

Gardner M (1970) The Fantastic Combinations of John Conway's New Solitaire Game 'Life'. Scientific American 223, 4, 120-123

Langton CG (1984) Self-reproduction in cellular automata. Physica D, 10,135–144

Macias NJ, Durbeck LJK (2002) Self-assembling circuits with autonomous fault handling. In A. Stoica, J. Lohn, R. Katz, D. Keymeulen, and R. S. Zebulum, editors, Proceedings of the 2002 NASA/DOD Workshop Conference on Evolvable Hardware, 46–55, Los Alamitos, CA, IEEE Computer Society Press

Mange D, Sipper M, Stauffer A, Tempesti G (2000) Toward robust integrated circuits: The Embryonics approach. Proceedings of the IEEE, 88, 4, 516–541

Mange D, StaufferA, Petraglio E, Tempesti G (2004a) Embryonic Machines That Divide and Differentiate. In A.J. Ijspeert, M. Murata, and N. Wakamiya, Eds., Biologically Inspired Approaches to Advanced Information Theory, Vol. 3141 of Lecture Notes in Computer Science, 201-216. Springer-Verlag, Berlin

Mange D, StaufferA, Petraglio E, Tempesti G (2004b) Self-Replicating Loop with Universal Construction Physica D, 191, 1-2, 178-192

Perrier JY, Sipper M, Zahnd J (1996) Toward a viable, self-reproducing universal computer. Physica D, 97,335–352

Pesavento U (1995) An implementation of von Neumann's self-reproducing machine. Artificial Life, 2, 4, 337-354

Reggia JA, Armentrout SL, Chou HH, Peng Y (1993) Simple systems that exhibit self-directed replication. Science, 259, 1282–1287

Restrepo HF, Mange D (2001) An embryonics implementation of a self-replicating universal Turing machine. In Liu Y, Tanaka K, Iwata M, Higuchi T, Yasunaga M editors (2001) Evolvable Systems: From Biology to Hardware (ICES 2001), volume 2210 of Lecture Notes in Computer Science, 74–87, Springer-Verlag, Berlin

Roco MC, Bainbridge WS editors (2002) Converging technologies for improving human performance. Nanotechnology, biotechnology, information technology and cognitive science. NSF/DOC - sponsored report, Arlington, VA

Sanchez E (1996) Field Programmable Gate Array (FPGA) Circuits. Towards Evolvable Hardware, Lecture Notes in Computer Science 1062, pages 1–18, Springer, Berlin

Stauffer A, Sipper M (2003) Data and signals: A new kind of cellular automaton for growing systems. In Lohn J, Zebulum R, Steincamp J, Keymeulen D, Stoica A, Ferguson MI editors (2003), Proceedings of the 2003 NASA/DOD Conference on Evolvable Hardware, pages 235–241, Los Alamitos, CA, IEEE Computer Society.

Tempesti G (1995) A new self-reproducing cellular automaton capable of construction and computation. In Morán F, Moreno A, Merelo JJ, Chacón P editors (1995) ECAL'95: Third European Conference on Artificial Life, volume 929 of Lecture Notes in Computer Science, pages 555–563, Heidelberg, Springer-Verlag

Tempesti G, Mange D, Stauffer A, Teuscher C (2002) The BioWall: An electronic tissue for prototyping bio-inspired systems. In Stoica A, Lohn J, Katz R, Keymeulen D, Zebulum RS editors (2002) Proceedings of the 2002 NASA/DOD Workshop Conference on Evolvable Hardware, pages 221–230, Los Alamitos, CA, IEEE Computer Society Press.

Trimberger S (1994) Field-Programmable Gate Array Technology. Kluwer Academic Publishers, Boston

von Neumann J (1966) Theory of Self-Reproducing Automata. University of Illinois Press, Illinois, Edited and completed by A. W. Burks

Watson JD, Crick FHC (1953) A structure for desoxyribose nucleid acid. Nature, 171, 737–738

Wolfram S (1994) Cellular Automata and Complexity. Addison-Wesley, Reading, MA

Part II

Prediction and Elucidation of Stream Ecosystems

Chapter 8

Development and Application of Predictive River Ecosystem Models Based on Classification Trees and Artificial Neural Networks

P. Goethals · A. Dedecker · W. Gabriels · N. De Pauw

8.1 Introduction

Prediction of freshwater organisms based on machine learning techniques is becoming more and more reliable due to the availability of appropriate datasets and modelling techniques. Artificial neural networks (Lek and Guegan 1999), fuzzy logic (Barros *et al*. 2000), evolutionary algorithms (Caldarelli *et al*. 1998), cellular automata (Gronewold and Sonnenschein 1998), etc. proved to be powerful tools to perform ecological modelling, especially when large datasets are involved. Models have several interesting applications in river management. They allow for a better interpretation of the results, easing the cause-allocation of the actual river status and increasing the insight needed to improve assessment systems (Fig. 8.1.). Models also allow for simulating the effect of potential management options and thus supporting decision-making. The development of effective and efficient monitoring networks based on models is probably another important advantage.

The 'River Invertebrate Prediction and Classification System' (RIVPACS) approach, based on statistical modelling, is currently one of the best available techniques for assessing the biological quality of running waters because it offers the ability to use environmental variables to predict species that are expected to occur at a site if it is unstressed. The expected fauna is then compared with the observed community of macroinvertebrates in order to assess the river quality (Wright *et al*. 2000). However, biological communities are dynamic and the nature of RIVPACS would need to be altered in order to predict a change in faunal composition in response to new environmental conditions at a given site (De Pauw 2000).

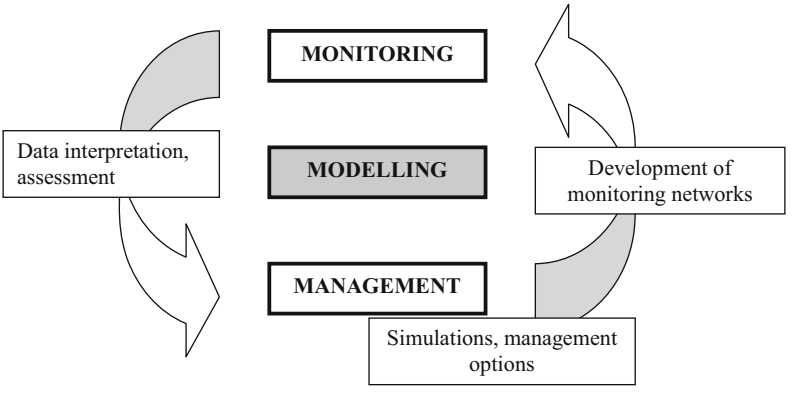


Fig. 8.1. Potential applications of ecosystem models in integrated river management (Goethals and De Pauw 2001).

In this context, models based on classification trees and artificial neural networks were developed and applied to predict the macro-invertebrate communities in the Zwalm river basin located in Flanders, Belgium. Because the macroinvertebrate communities have drastically changed as a result of several types of human impacts, it was attempted to train the different models in order to predict the effect of these different types of environmental stress.

8.2
Study Sites, Data Sources and Modelling Techniques

8.2.1
The Zwalm River Basin

The Zwalm river basin is part of the Scheldt river basin (Carchon and De Pauw 1997). The Zwalm river basin has a total surface of 11,650 ha. The Zwalm river itself has a length of 22 km (Fig. 8.2.). The average water flow at Nederzwalm, very near the Scheldt is about one m^3s^{-1}. It has a very irregular regime, with low values in the summer (minima lower than 0.3 m^3s^{-1}) and relatively high values in rainy periods (maxima up to 4.7 m^3s^{-1}) (Lauryssen *et al*. 1994). The water quality in the Zwalm river basin substantially improved during the year 1999 due to investments in sewerage and wastewater treatment plants during the last years (VMM, 2000). None the less, most parts of the river are still polluted by untreated urban wastewater discharges and by diffuse pollution originating from agricultural activities.

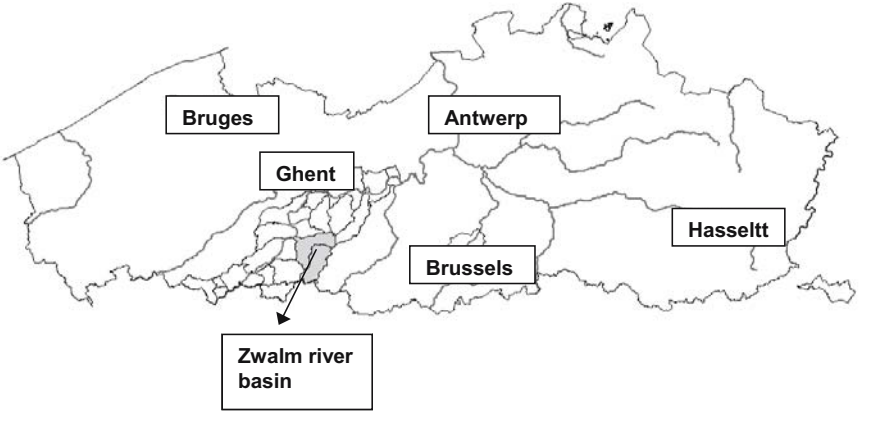


Fig. 8.2. The Zwalm river basin in Flanders (Belgium).

Although Flanders is in general rather flat, the Zwalm river basin is characterized by several height differences, resulting in a very unique river ecosystem within the Flemish region (Soresma 2000). Consequently, soil erosion is the most important geo-morphological process resulting in a substantial transport of (contaminated) sediments in the river (AMINAL 1999). Also structural and morphological disturbances are numerous (Carchon and De Pauw 1997). Weirs for water quantity control obstruct fish migration and are one of main ecological problems within the river basin. Therefore an in-depth study has been made on the construction of fish migration channels and also natural overflow systems to reach an ecologically friendly water quantity management in the near future (Soresma 2000). Some upper parts of the watercourses in the Zwalm river basin are colonized by very rare species as Bullhead (*Cottus gobio*), Brook Lamprey (*Lampetra planeri*) and several vulnerable macro-invertebrates.

8.2.2 Data Collection

Structural characteristics (meandering, substrate type, flow velocity, …) and physical-chemical variables (dissolved oxygen, pH, …) were used as inputs to predict the presence or absence of macroinvertebrate taxa in the headwaters and brooks of the Zwalm river basin (see Table 8.1.). Structural characteristics were visually monitored (Dedecker 2001). Flow velocity was determined by timing the transport of a float over a distance of 10 m. Field measurements were made for temperature and dissolved oxygen (TW OXI 330/SET), pH (Jenway 071) and conductivity (WTW LF 90). Suspended solids were measured in the laboratory based on spectrophotometric measurements (Dedecker 2001).

Table 8.1. Monitored variables in the Zwalm river basin.

Variables	Units
pH	
Temperature	°C
Dissolved oxygen	mg/l
Conductivity	µS/cm
Suspended solids	mg/l
Water level	cm
Fraction of pebbles	%
Shadow	%
Water plants	2 classes: 0 = absent; 1 = present
Width	cm
Flow velocity	m/s
Meandering	6 classes (1 = well developed to 6 = absent)
Hollow river beds	6 classes (1 = well developed to 6 = absent)
Pools/Riffles	6 classes (1 = well developed to 6 = absent)
Artificial embankment structures	3 classes (0 = absent; 1 = moderate; 2 = intensive)

Macroinvertebrates were collected by means of a standard handnet (NBN 1984) during five minute kick sampling. The objective of the sampling consists in collecting the most representive diversity of the macroinvertebrates within the examined site (De Pauw and Vanhooren 1983). The absence or presence of macroinvertebrate taxa was respectively represented by 0 or 1 for use in the different models. In total, 60 sites were monitored in the Zwalm river basin.

8.2.3 Classification Trees

Classification trees (Breiman *et al*. 1984), often referred to as decision trees (Quinlan 1986) predict the value of a discrete dependent variable with a finite set of values (called class) from the values of a set of independent variables (called attributes), which may be either continuous or discrete. Data describing a real system, represented in the form of a table, can be used to learn or automatically construct a decision tree.

The common way to induce decision trees is the so-called 'Top-Down Induction of Decision Trees' (TDIDT, Quinlan 1986). Tree construction proceeds recursively starting with the entire set of training examples. At each step, the most informative attribute is selected as the root of the (sub)tree and the current training set is split into subsets according to the values of the selected attribute. For discrete attributes, a branch of the tree is typically created for each possible value

of the attribute. For continuous attributes, a threshold is selected and two branches are created based on that threshold. For the subsets of training examples in each branch, the tree construction algorithm is called recursively. Tree construction stops when all examples in a node are of the same class (or if some other stopping criterion is satisfied). Such nodes are called leaves and are labelled with the corresponding values of the class.

A number of systems exist for inducing classification trees from examples, e.g., CART (Breiman *et al*. 1984), ASSISTANT (Cestnik *et al*. 1987), and C4.5 (Quinlan 1993). Of these, C4.5 is one of the most well known and widely-used decision tree induction systems. J48 (Witten and Frank 1999) is a Java re-implementation of C4.5. It is a part of the machine learning package WEKA, which also includes some of the latest developments in machine learning. This J48 was also used for inducing classification trees and prediction models of macroinvertebrate taxa of the Zwalm river basin. Standard settings were used. The model validation was based on splitting the dataset in a training and validation set of respectively 40 and 20 instances. Also ten fold cross validation on the whole dataset (60 instances) was applied in specific cases.

8.2.4 Artificial Neural Networks

Artificial neural networks (ANNs) are mathematical models based on the transfer of information through a network of functional units, called neurons. Given a number of input values, entered at the basis of the network, it generates one or more outputs. ANNs are currently recognized as an alternative for multivariate statistics to predict aquatic communities (Gabriels *et al.* 2000). Recently, several studies have been published, concerning the application of neural networks for relating freshwater macroinvertebrates with their abiotic environment (e.g. Walley and Fontama 1998; Schleiter *et al*. 1999; Gabriels *et al.* 2000). The neural network was implemented with the neural network extension of the software package MATLAB 5.3 for MS Windows™ (Demuth and Beale 1998). The model validation was based on splitting the dataset in a training and validation set of respectively 40 and 20 instances. Also 10 fold cross validation was sometimes applied, as described by Witten and Frank (2000). Several optimisation studies were carried out to select the best model configuration (Dedecker 2001). The best neural network consisted of one hidden layer and ten neurons, with ‘tansig’ and ‘logsig’ transfer functions and ‘gradient descending with momentum and adaptive learning rate backpropagation’ as training algorithm (Demuth and Beale 1998). A scheme of this neural network is shown in Fig. 8.3. During the research, special interest was paid to the influence of the frequency of occurrence of the taxa on the prediction reliability of the developed models. Sensitivity analyses were used to get insight in the applied ‘concepts’ of these black box models.

8.2.5
Model Assessment

To compare different models, a validation set of 20 instances was submitted as input to the models and the predicted outputs were then compared to the measured results. In this way, the amount of 'correctly classified instances' (CCI) was obtained and could be used to compare the performance of the different modelling techniques. Ten fold cross validation (Witten and Frank 2000) for training and validation on the whole dataset was used in some specific cases and is explicitly mentioned in the text.

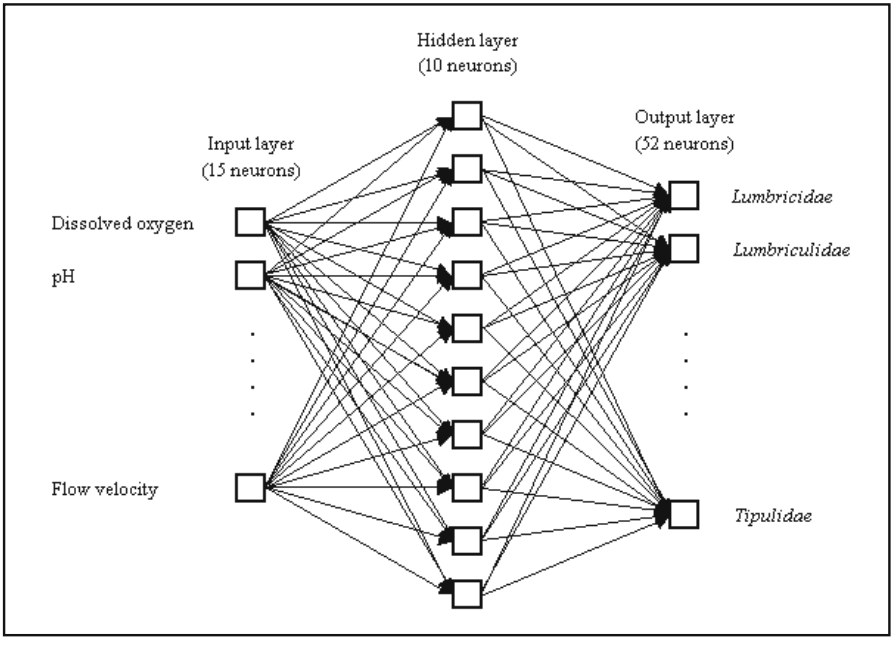


Fig. 8.3. Scheme of the applied neural networks for the Zwalm river data.

8.3 Results

8.3.1 Classification Trees

8.3.1.1 Model Development and Validation

Classification trees were constructed for all 52 taxa collected during the 60 samplings in the headwaters of the Zwalm river basin. The reliability of the predictions differs dramatically between the macroinvertebrate taxa. The frequency of occurrence of the taxa in the different sites is one of the major explanations of this phenomenon (Table 8.2). Especially when the taxa are very common or extremely rare, the amount of correctly classified instances is very high during the validation process, but this can mainly be explained by the high reliability to make a good prediction merely based on a probabilistic guess. The J48 does not induce a meaningful tree in these cases, as can be seen in Table 8.2. for *Aplexa* and *Tubificidae*.

Table 8.2. Prediction of three different macroinvertebrate taxa by means of classification trees (CCI calculation based on tenfold cross validation, the database consisted of 48 instances).

Taxa	Frequency of occurrence in the Zwalm (%)	Correctly Classified Instances (%)	Number of variables in the model	Number of leafs (model complexity)
Aplexa	2	100	0	1
Asellidae	43	63	2	3
Tubificidae	93	94	0	1

The J48 algorithm is mainly interesting for moderately frequent taxa, such as *Asellidae* and *Gammaridae* (Fig. 8.4.). Based on tenfold cross validation, the CCI score is 63 % for predicting *Gammaridae*. The tree also reveals interesting information concerning the variables that are important to predict this taxon. The main variables for the prediction of *Gammaridae* are water level, amount of hollow river beds, amount of stones, dissolved oxygen and pH. From the values in the leafs of the tree one can conclude that the *Gammaridae* mainly prefer the upstream parts of the river basin. The taxon is present in undeep waters (water level lower than 10.5 cm). It also prefers hollow beds and cavities, which nearly

only occur in fast running waters, thus also the higher and steeper upstream parts. *Gammaridae* also prefer more stony material, this means quickly running streams where sediments do not settle. In cases of deeper waters without cavities (due to artificial embankments) the *Gammaridae* are only present when the dissolved oxygen percentage is sufficiently high. The pH value plays a role under specific circumstances, but this classification is of the lowest importance in the tree.

8.3.1.2 Application of Predictive Classification Trees for River Management

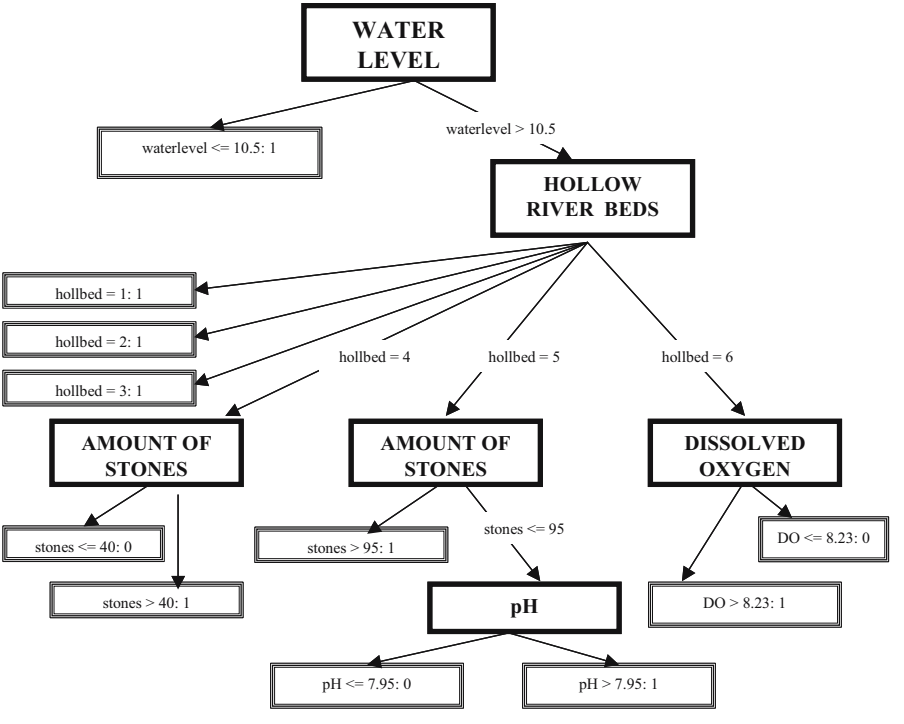


Fig. 8.4. Example of a classification tree model of *Gammaridae* in the Zwalm river basin. The single bold frames contain the classification variables, while the triple frames contain the final prediction of the *Gammaridae* (0 = absent and 1 = present). Hollbed is an ordinal categorical variable (six classes: 1= very good hollow beds under trees; 2 = good hollow beds; 3 = hollow beds by erosion under vegetation; 4 = moderate cavities; 5 = hollow beds not probable; 6 = no hollow beds by artificial embankments).

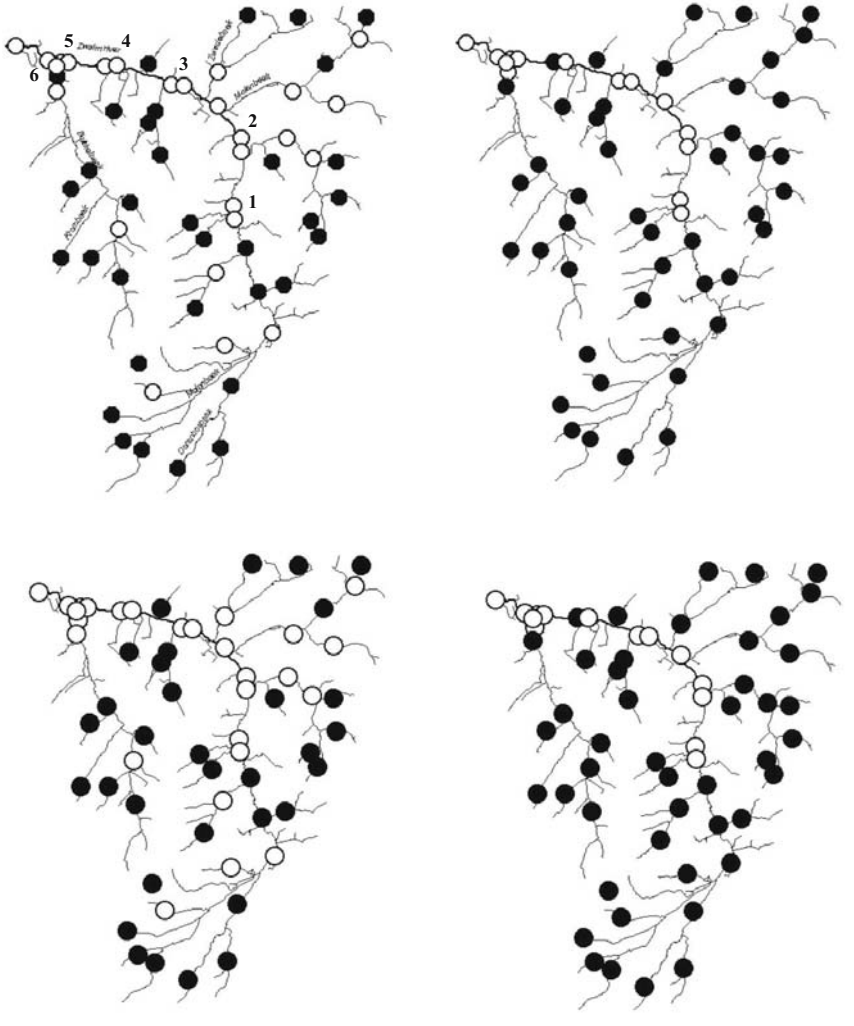


Fig. 8.5. Distribution plots of *Asellidae* in the Zwalm river basin, black spots indicate the absence of *Asellidae*, while white spots the presence. The four maps give a representation of what effect the removal of six weirs (1, 2, 3, 4, 4 and 6) in the Zwalm river basin can have on the *Asellidae* populations (top left = measurements August/September 2001; top right = J48 classification tree simulations (for August/September 2001); bottom left = simple prediction of the weirs-removal impact, the site in front of the weir will have the same characteristics as behind the weir before weir removal, while the situation behind

the weir is not altered; bottom right = J48 classification tree simulations of weirs-removal).

In Fig. 8.5. the effect on the Asellidae population of the removal of the 6 weirs in the Zwalm river basin is simulated by classification trees. If the two upper maps are compared, one can quickly notice the generalisation the model makes: according to the model the *Asselidae* only colonize the broader river sites (the only rule generated by the J48-model using tenfold cross validation on all sixty instances is:'width more than 3.5 meters: *Asellidae* present, while absent in the more narrow streams').

Althought the amount of correctly classified instances is rather good and the induced model gives an interesting generalisation to easily and reliably predict *Asellidae* in the field, the model is not interesting to predict the effects of the removal of the weirs. The maps at the bottom illustrate that the *Asellidae* will keep on colonizing the river stretches in sites 1 to 6 after the removal of the six weirs, according to the J48 model. When a simple rule ('the site in front of the weir will have the same characteristics as behind the weir before weir removal, while the situation behind the weir is not altered') is used to predict the *Asellidae* behaviour, similar predictions are made. According to ecological experts however, the *Asellidae* do not effectively colonize the sites behind the weir as could be thought according to the measurements. Most probably the presence of the *Asselidae* in the samples behind the weirs has to be explained by accidental carry-away processes from the sites in front of the weirs, where the conditions (slow current) are convenient for the *Asselidae*. This also explains that in the samples of the river stretches some kilometres in front of the weirs, *Asellidae* are never present. So according to ecological experts, the *Asselidae* will most probably not colonize the Zwalm river under undisturbed conditions and also not when the weirs are removed.

8.3.2 Artificial Neural Networks

8.3.2.1 Model Development and Validation

From Fig. 8.6. one can clearly observe that artificial neural networks (ANNs) make better predictions compared to simple probabilistic guesses. Another trend, similar to the predictions with classification trees is that the reliability of the models is the highest for very common (*Chironomidae*, *Tubificidae*) and extremely rare taxa (*Aplexa*, *Ephemera*, *Armiger*). The added value of the artificial neural network is the lowest under these circumstances.

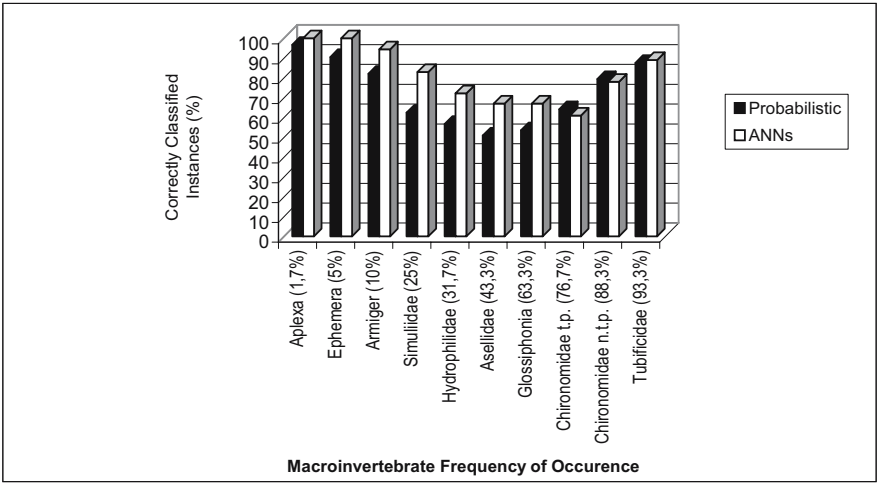


Fig. 8.6. Prediction of macroinvertebrate taxa in the Zwalm river basin based on ANNs compared to simple probabilistic guesses.

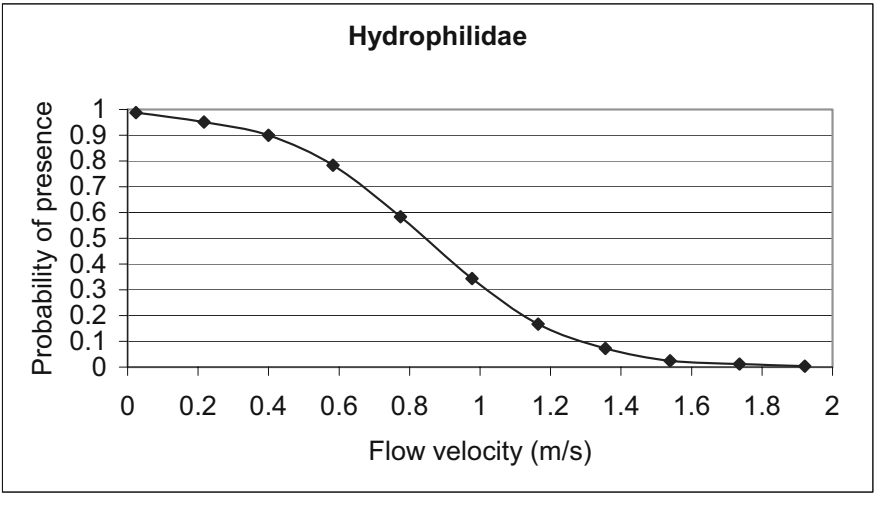


Fig. 8.7. Probability of presence predicted by the ANN for *Hydrophilidae* in relation to flow velocity.

To get insight in the inference system of the ANN models, all input variables except the one of interest are kept constant (at the average value of the database). In this way, one is able to determine the impact of the variable on the presence or absence of a specific taxon. From Fig. 8.7. one can conclude that *Hydrophylidae* prefer rather slowly running or stagnant waters, while *Gammaridae* rather prefer fast running waters (Fig. 8.8.).

In this way, some insight is gained in the habitat preference of all taxa, what delivers substantial information for river ecosystem management.

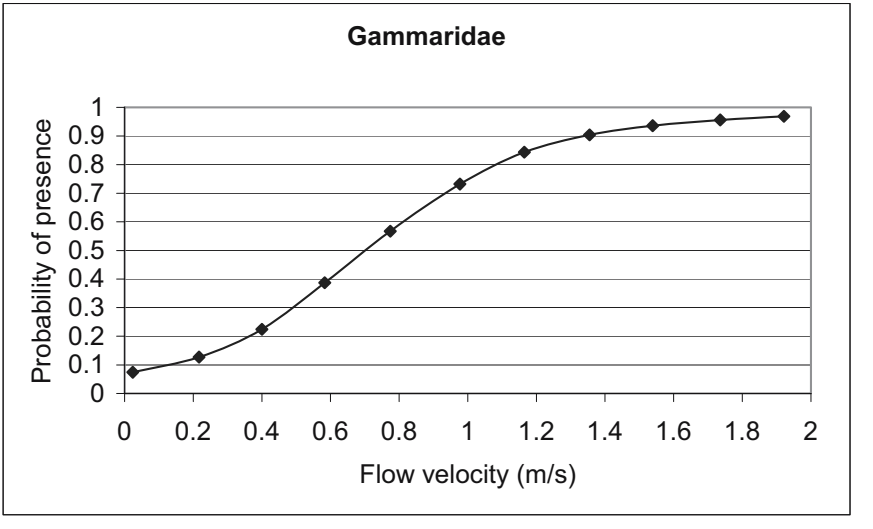


Fig. 8.8. Probability of presence predicted by the ANN for *Gammaridae* in relation to flow velocity.

8.3.2.2 Application of Predictive Artificial Neural Networks for River Management

8.3.2.2.1 Prediction of Environmental Standards

In Fig. 8.9. the convenience of ANN models for determining ecotoxicological information is presented. Using these ecotoxicty-curves developed by ANN model simulations, one can define environmental standards on a data driven basis. In Fig. 8.9. a 90% protection level for *Gammaridae* is shown for dissolved oxygen (7%).

By developing such curves for all types of taxa, the environmental standards can be defined on a more scientific basis than is nowadays often done.

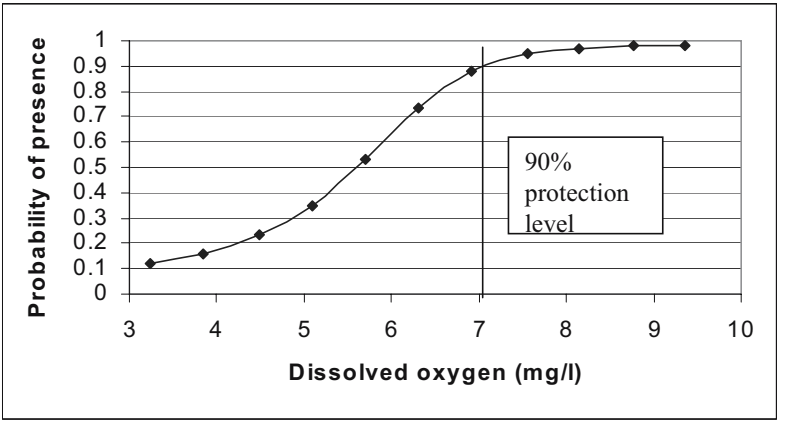


Fig. 8.9. The impact of dissolved oxygen on the probability of presence of *Gammaridae* and the 90% protection level (7% of dissolved oxygen).

8.3.2.2.2 Feasibility Analysis of River Restoration Options

An illustration about the application of ANN models to mitigate human impacts on the Bettelhovebeek is presented in this case study. Several structural modifications are dramatically affecting the biological communities at this site.

Tab. 8.3. Optimal restoration option for the Bettelhovebeek and predicted macroinvertebrate taxa and BBI after river restoration.

Structural characteristics	**Before river restoration Actual bad situation**	**After river restoration Future good status**
Artificial embankments	concrete and iron str.	none
Meandering	none	moderate
Deep/shallow variation	none	moderate
Hollow river beds	none	well developed
Macroinvertebrate taxa	**Actual bad status (field measurement)**	**Future good status (ANN prediction)**
Sialis	absent	present
Limnephilidae	absent	present
Simuliidae	absent	present
Belgian Biotic Index	**moderate quality (BBI=5)**	**good quality (BBI=7)**

The aim of this study was to determine the most efficient restoration option to obtain a stable biological ecosystem meeting the minimal river water quality standards for Flanders, such as 'the Belgian Biotic Index (BBI) equal or higher than seven'. In Tab. 8.3 only predictions for the best restoration option are summarized, based on a selection made from a set of simulations with artificial neural network models (Dedecker 2001). The results indicate that after river restoration, some macroinvertebrate taxa, indicative for a good water quality and that are currently not present, will colonize the site again. Also the predicted BBI changes from a moderate to a good quality and illustrates that the basic water quality standards for Flanders are met under the mitigated conditions.

8.4 Discussion

Generally the classification trees performed well to predict the macroinvertebrate taxa, based on the fifteen input variables. This method does not merely generate results with a low prediction error, but also allows the user to identify associations and general trends in the data (as illustrated by the *Gammaridae* model), making it more interesting than complete black-box techniques. One may conclude that classification trees are interesting grey-box prediction techniques, allowing the user to combine a small prediction error with getting some information on general trends in the data. This methodology can thus be used to determine the ecological requirements of organisms that are not sufficiently well understood (Dzeroski *et al*. 1997). Probably the results can be improved by providing other valuable inputs. Experiments with different sets of input variables did not only result in an altered prediction error, but also the complexity of the trees as well as the relative importance of some 'general' trends seemed to be affected. Further research is therefore necessary to get insight in the impact of different input variable sets on the prediction qualities of decision trees. D'heygere *et al.* (2001) illustrated the convenience of genetic algorithms to automatically select input variables sets in this context. Also models for very common or very rare species need to be optimised. The trees of these taxa are very limited and have in many cases no added value to black box models and probabilistic guesses. Boosting, bagging and meta-cost algorithms (Witten and Frank 2000) can have an interesting added value for this, although in many cases the transparency and robustness as well as their ecological validity of the induced rules is often affected by these techniques.

Although rule-induction by classification trees generates in general robust models with a high predictive reliability, one has to be aware of the static characteristics of this type of models. Therefore, the simulations still need to be checked by ecological experts that can deliver knowledge that is often not included in the database used for the model induction. This was illustrated by the case study on the weirs-removal in the Zwalm river basin. To increase the model feasibility with regard to simulations for river restoration management, spatial-temporal expert-rules will have to be included (such as migration kinetics of the

organisms in water, land and air), as well as annual measurement campaigns that can improve the database with regard to its information contents.

The performance of the ANN models is in general better than simple probabilistic predictions and rather similar to classification tree models. Also Walley and Fontama (1998), Schleiter *et al*. (1999) and Gabriels (2000) came to similar conclusions for predicting macroinvertebrates based on a limited set of environmental characteristics. ANN models for common and rare taxa have the highest reliability (expressed as correctly classified instances or CCI), while for moderately frequent taxa the prediction is lower, but relatively much better than the probabilistic guesses as was also reflected in the classification tree models. Sensitivity analyses allowed to study the impact of the input variables on the presence or absence of macroinvertebrate taxa. The impact of flow velocity on the absence and presence of *Gammaridae* is confirmed by earlier observations (De Pauw and Vannevel 1993) and also by the rules induced via the classification tree models. Many other relations were detected and were in most cases confirmed with related ecological research results, when this information was available. This also indicates that these models in many cases work in an ecological meaningful manner. In this way ANN models allow to determine the major variables that affect the ecosystem quality and should be taken under direct consideration in the river ecosystem management. Further research is necessary to determine the optimal neural network configuration. Walley and Fontama (2000) used an ANN with two hidden layers with six nodes each for similar simulations. Also the impact of the applied training algorithms as well as the risk of overtraining the network should be further analysed to obtain reliable and meaningful predictions in the long run.

Several case studies related to restoration (e.g. Bettelhovebeek) and environmental impact assessment proved the interesting added value of the ANN ecosystem models developed for river management.

Acknowledgements

The authors like to thank the Scientific Research Foundation of Flanders (FWO-Flanders) for its financial support (project 3G01.02.97). Special thanks to Nico Raes for his support with regard to the data collection and to Saso Dzeroski for his help on the classification tree induction.

References

AMINAL (1999) Control of sediment transport in unnavigable watercourses as part of integrated water management: Zwalm river basin project. AMINAL, Ghent (in Dutch)

Barros LC, Bassanezi RC, Tonelli PA (2000) Fuzzy modelling in population dynamics. Ecological Modelling, 128, 27-33

Breimann L, Friedman JH, Olshen RA, Stone CJ (1984) Classification and regression trees. Pacific Grove,Wadsworth

Caldarelli G, Higgs PG, McKane AJ (1998) Modelling coevolution in multispecies communities. Journal of Theoretical Biology, 193, 345-358

Carchon P, De Pauw N (1997) Development of a methodology for the assessment of surface waters. Study by order of the Flemish Environmental Agency (VMM). Ghent University, Laboratory of Environmental Toxicology and Aquatic Ecology, Gent, 55p (in Dutch)

Cestnik B, Kononenko I, Bratko I (1987) ASSISTANT 86: A knowledge elicitation tool for sophisticated users. In: Bratko, I. & Lavrak, N. Progress in machine learning. Sigma Press, Wilmslow

Dedecker A (2001) Modelling of macroinvertebrate communities in the Zwalm River basin by means of artificial neural networks. M. Eng. thesis, Ghent University, Faculty of Applied Biological Sciences, in preparation (in Dutch)

D'Heygere T, Goethals P, De Pauw N (2001) Application of genetic algorithms for input variables selection of decision tree models predicting mollusca in unnavigable Flemish watercourses. Mededelingen Faculteit Landbouwkundige en Toegepaste Biologische Wetenschappen, Ghent University, Gent, Belgium (in press)

Demuth H, Beale M (1998) Neural Network Toolbox for use with MATLAB. User's guide. Version 3.0. The Mathworks, Inc., Natick, USA

De Pauw N (2000) Using RIVPACS as a modeling tool to predict the impacts of environmental changes. p. 311-314. In: Wright, J.F., Sutcliffe, D.W., & Furse, M.T. Assessing the biological quality of fresh waters: RIVPACS and other techniques. FBA, Ambleside (UK)

De Pauw N, Vanhooren G (1983) Method for biological assessment of watercourses in Belgium. Hydrobiologia, 100, 153-168

De Pauw N, Vannevel R (1993) Macroinvertebrates and water quality. Dossiers Stichting Leefmilieu 11: Stichting Leefmilieu, Antwerp (in Dutch)

Dzeroski S, Grbovic J, Walley WJ (1997) Machine learning applications in biological classification of river water quality, p. 429-448. In: Michalski, R.S., Bratko, I. & Kubat, M. Machine learning and data mining: methods and applications. John Wiley & Sons Ltd., New York

Gabriels W, Goethals PLM, Heylen S, De Cooman W, De Pauw N (2000) Modelling benthic macro-invertebrate communities in Flanders using artificial neural networks. IWA, Proc. Watermatex 2000 Symposium, Ghent. p. 1.143-1.146

Goethals P, De Pauw N (2001) Development of a concept for integrated ecological river assessment in Flanders, Belgium. Journal of Limnology (in press)

Goethals P, Dzeroski S, Vanrolleghem P, De Pauw N (2001) Prediction of benthic macroinvetebrate taxa (Asellidae and Tubificidae) in watercourses of Flanders by means of classification trees. Paper submitted at the IWA 2nd World Water Congress Berlin 2001

Gronewold A, Sonnenschein M (1998) Event-based modelling of ecological systems with asynchronous cellular automata. Ecological Modelling, 108, 37-52

Lauryssen F, Tack F, Verloo M (1994) Nitrogen transport in the Zwalm river basin. Water, 75, 46-49 (in Dutch)

Lek S, Guegan JF (1999) Artificial neural networks as a tool in ecological modelling, an introduction. Ecological Modelling, 120, 65-73

NBN (1984) Biological water quality: determination of the biotic index based on aquatic macroinvertebrates, NBN T92-402. Institut Belge de Normalisation (IBN) (in Dutch and French)

Quinlan JR (1986) Induction of decision trees. Machine Learning, 1(1), 81-106

Quinlan JR (1993) C4.5 : Programs for machine learning. Morgan Kaufmann, San Francisco

Schleiter IM, Borchardt D, Wagner R, Dapper T, Schmidt KD, Schmidt HH, Werner H (1999) Modelling water quality, bioindication and population dynamics in lotic ecosystems using neural networks. Ecological Modelling, 120, 271-286

Soresma (2000) Environmental impact assessment report on the development of fish migration channels and natural overflow systems in the Zwalm river basin. Soresma Advies- en Ingenieursbureau, Antwerp (in Dutch)

VMM (2000) Water quality - water discharges 1999. Flemish Environmental Agency, VMM, Erembodemgem (in Dutch)

Walley WJ, Fontama VN (1998) Neural network predictors of average score per taxon and number of families at unpolluted sites in Great Britain. - Water Research, 32, 613-622

Walley WJ, Fontama VN (2000) New approaches to river quality classification based upon artificial intelligence. p. 263-280. In: Wright, J.F., Sutcliffe, D.W., & Furse, M.T. Assessing the biological quality of fresh waters: RIVPACS and other techniques. FBA, Ambleside (UK)

Witten IH, Frank E (2000) Data mining: practical machine learning tooals and techniques with Java implementations. Morgan Kaufmann Publishers, San Francisco

Wright JF, Sutcliffe DW, Furse MT (2000) Assessing the biological quality of fresh waters: RIVPACS and other techniques. FBA, Ambleside (UK). pp 373

Chapter 9

Modelling Ecological Interrelations in Running Water Ecosystems with Artificial Neural Networks

I.M. Schleiter · M. Obach · R. Wagner · H. Werner · H.-H. Schmidt · D. Borchardt

9.1 Introduction

The assessment of properties and processes in running water ecosystems is a major issue in basic and applied aquatic science and has consequences for environmental management. However, knowledge of the system functions, e.g. temporal and spatial dynamics of physical, chemical, hydro-morphological and biological processes, and species-habitat interrelations are still insufficient. An integrative and prognostic ecological assessment of running waters thus is presently not available (e.g. Bayerisches Landesamt für Wasserwirtschaft 1998; Resh et al. 1994; Statzner et al. 1994; Townsend and Hildrew 1994; Townsend 1989; Vannote et al. 1980).

The analysis of running water ecosystems and prediction with deterministic and stochastic models are limited. However, studies on water quality assessment have improved the methodology, which can also be applied to basic science.

The high complexity and the spatial and temporal system dynamics are examples of typical non-linear relationships of abiotic and biotic variables with often low amounts of non-normally distributed data. This limits the application of traditional statistics. Artificial neural networks (ANNs) provide an alternative tool to analyse and model ecological relationships. Their most important features are multi-dimensionality, non-linearity, the ability to learn from examples and to generalise.

General aims of our modelling approach are:

Application and development of ANNs to

1. visualise and test data reliability
2. model ecological relations
3. test the suitability of different ANN types and pre-processing methods
4. detect the most important input variables
5. visualise the network status and the activation of neurons
6. develop neural modelling techniques initiating further research on bioindication and ecological prediction

In basic and applied running water ecology detection and description of unknown interrelations and generation of hypotheses identification of the most relevant

variables for modelling, e.g. indicator species prediction of ecological properties of lotic ecosystems prediction of the assemblage of benthic communities in disturbed and undisturbed streams generalisation of interdependencies.

In an interdisciplinary research project of mathematicians, computer scientists, ecologists, and engineers, the suitability of various types of ANNs was tested. They were used to model temporal dynamics of water quality based on weather, urban storm-water run-off and waste-water effluents, bioindication of lotic ecosystem properties using benthic macroinvertebrates, and long-term population dynamics of aquatic insects depending on environmental and ecological variables.

9.2 Materials and Methods

9.2.1 Data Base

Our research was based on two data sets from running waters in Hesse (Central Germany):

Nine streams with different amounts of organic pollution (Schleiter et al. 1999; 2001) (248 macro-zoobenthos taxa and physical, chemical, and hydro-morphological variables)

A thirty years data set of environmental variables (precipitation, discharge, water temperature) and aquatic insects (Ephemeroptera, Plecoptera, Trichoptera) of an almost pristine stream (Limnological River Station Schlitz; Obach et al. 2001).

9.2.2 Data Pre-Processing

Pre-processing includes all data alterations before the applications of ANNs. The first step is a test of completeness, e.g. determination of an adequate method to handle missing values, and plausibility. Outliers can be detected and mapped onto the borders of a reliable range of values (truncation) or use non-linear functions, e.g. logarithmic or sigmoidal.

Variables can be normalized in order to avoid an undesireably high influence of large absolute values. We usually mapped the values linearly onto the interval [0,1]; occasionally standardisation is more advisable.

The data set was usually divided into training (adaption of net parameters), verification (selection of an adequate model) and test data (to estimate generalisation ability).

Selection of the most relevant variables by regression, correlation analyses or based on expert knowledge or combination of variables by Principal Component

Analysis (PCA) leads to reduced computational efforts and easier to manage models.

9.2.3
Artificial Neural Network Types

The software for all applied ANN simulations was developed by the Research Group Neural Networks at the University Kassel (Germany). The suitability of the following ANN types was tested: Kohonens self organising map as an example for unsupervised learning networks and feed-forward networks such as conventional and modified Multi-Layer-Perceptrons (MLP, Rumelhart et al. 1986; Senso Neural Networks, Dapper 1998), General Regression Neural Networks (GRNN, Specht 1991), Motoric Feature Maps (MFM, Ritter et al. 1994), Linear Neural Networks (LNN) and also Radial Basis Function Neural Networks (RBF, Bishop 1995).

9.2.4
Dimension Reduction

Ecological variables are often correlated and thus data contain a certain amount of redundant information. For example, many variables in aquatic environments depend directly or indirectly on the amount of oxygen available. Dimension reduction finally resulted in:

1. an improvement of the proportion of the number of input dimensions and the amount of available data with an improved generalization quality of ANNs
2. reduced computation effort, particularly during ANNs training
3. recognition of relevant predictors (to be compared also with expert knowledge)
4. simplified and easy to analyse models based on a manageable number of variables.

Data compression by linear PCA factor analysis and bottle-neck nets (Dapper 1998; Bishop 1995) provided independent variables. The computed factor "pollution with organic matter" contains compressed information on P_{tot}, O_2, BOD_5, COD, and is interpreted as a major stressor in many anthropogenically altered aquatic ecosystems. In contrast, selection of relevant predictors by sensitivity analysis (Dapper 1998), stepwise methods, genetic algorithms (Goldberg 1989) simply reduces the number of relevant variables to neglect irrelevant and redundant information.

9.2.5
Quality Measures

Model accuracy is usually measured as the differences between predicted and observed data. Usual error measures are the sum of squared errors (SQE), the

mean squared error (MSE), the root mean squared error (RMSE), and the coefficient of determination (B) (Bishop 1995).

Finer quality measures are provided below.

9.3 Data Exploration with Unsupervised Learning Systems

An initial application of ANNs in the process of data exploration is the use of unsupervised trained networks to visualise similarity of patterns, grouping of objects, and again outlier detection, but in multiple dimensions (Chon et al. 1996; Foody 1999). One of the most famous techniques is Kohonen's self organised map, SOM (Kohonen 1995).

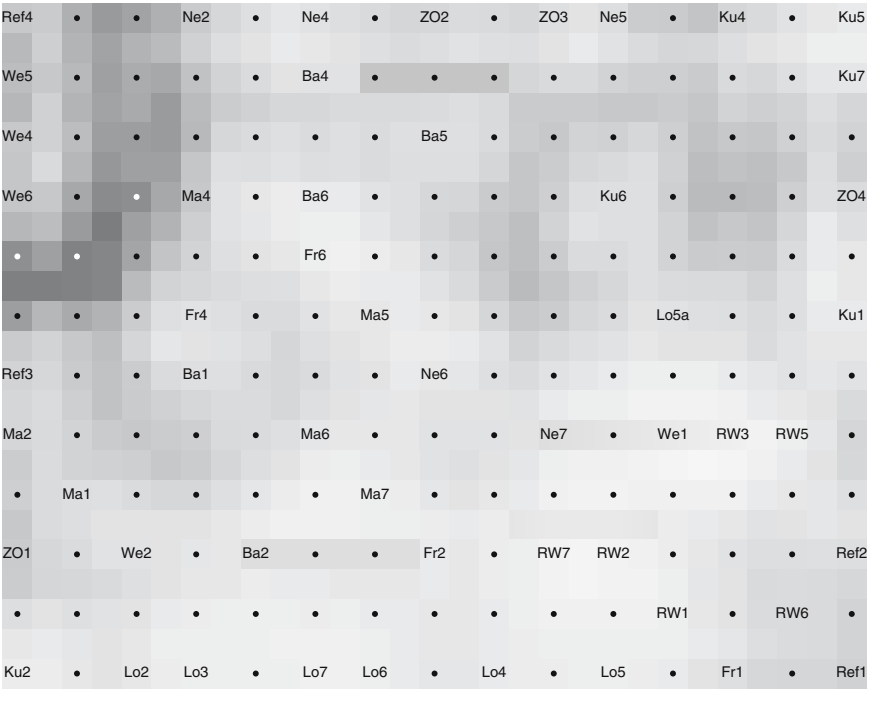

Ref4 Ne2 Ne4 ZO2 ZO3 Ne5 Ku4 Ku5
We5 Ba4 Ku7
We4 Ba5
We6 Ma4 Ba6 Ku6 ZO4
Fr6
Fr4 Ma5 Lo5a Ku1
Ref3 Ba1 Ne6
Ma2 Ma6 Ne7 We1 RW3 RW5
Ma1 Ma7
ZO1 We2 Ba2 Fr2 RW7 RW2 Ref2
RW1 RW6
Ku2 Lo2 Lo3 Lo7 Lo6 Lo4 Lo5 Fr1 Ref1

Figure 9.1. Similarities of the 55 samples with regard to eight water properties (water temperature, oxygen, conductivity, total nitrogen, total phosphorus, hydromorphological quality class, substrate diversity, number of species per sampling site) visualised by a U-matrix of a 15×12 SOM (spatial proximity and light grey shades mean similarity).

We usually used this tool, but provide only one example here. Similarities between sampling and reference sites of different streams were visualised with respect to water quality, habitat features and also by colonisation patterns of benthic macro-invertebrates and adult aquatic insects (Fig. 9.1.). Reference sites are a theoretical construct based on expert knowledge of typical chemical and hydro-morphological quality classes: unpolluted and natural (Ref. 1), moderately polluted and modified (Ref. 2), heavily polluted and modified (Ref. 3), extremely polluted and completely degraded (Ref. 4) (LAWA 1998; Hessisches Ministerium für Umwelt, Landwirtschaft und Forsten 2000).

The U-matrix (Fig. 9.1.) (Ultsch 1993) of the SOM visualises the similarities between samples by location and shading of the space between the neighbouring codebook vectors, i.e. prototypes. More than one sampling site may be mapped on the same codebook vector, e.g. Ku2 and Ku3. Small distances between codebook vectors and light grey shades indicate similar chemical and hydro-morphological quality. For instance, the samples We 4, 5, 6 and the reference Ref. 4 (top left) are clearlyseparated by their marginal position and dark border. The other samples show no clear separation, but rather transient regions. As expected, the most contrasting references Ref. 1 and 4 take opposite locations.

Trajectories of subsequent sampling sites along streams can be visualised within a SOM (Fig 9.2.). Some streams, particularly RW, show only small differences between samples, whereas others vary considerably. For example, Ku2 and Ku3 are mapped onto the same codebook vector, their large distances to Ku1 and to Ku4 indicate environmental impact. A fishpond between Ku1 and Ku2 and a storm water overflow between Ku2 and Ku3 caused damages, mainly to the hydro-morphological status (see plane hydro-morphological quality class). Two additional storm water basins between Ku3 and Ku4 affected a further decrease of the water quality, whereas the hydro-morphological quality improved. The waste water treatment plant between Ku4 and Ku5 caused no change of the water properties. Between Ku5 and Ku6, river bed morphology deteriorated whereas water quality remained unaffected. Both recuperated downstream Ku6. Another option to visualise the similarities of the samples is the *Sammon map* (Sammon 1969) of the SOM codebook vectors (Fig. 9.3.), approximately preserving their distances.

The values of the components of the codebook vectors (water or hydro-morphological quality) can be visualised by grey levels on the SOM – the darker the grey shade, the lower the value (Fig. 9.4.). The interrelations of variables become evident, e.g. the oxygen and the water temperature planes are complementary.

In conclusion, SOMs are an excellent exploration tool for multidimensional variables, providing visual aids for inspecting unknown data, outlier detection, and initial grouping of data. In contrast to many other cluster analysis methods, SOMs also handle data with smooth transitions, which are often typical for ecological data (e.g. Vannote et al. 1980). The visualisation capability (grey shades and distances) of SOMs shows similarities between objects and groups of objects. This may inspire hypothesis generation and analysis of hybrid networks (see below).

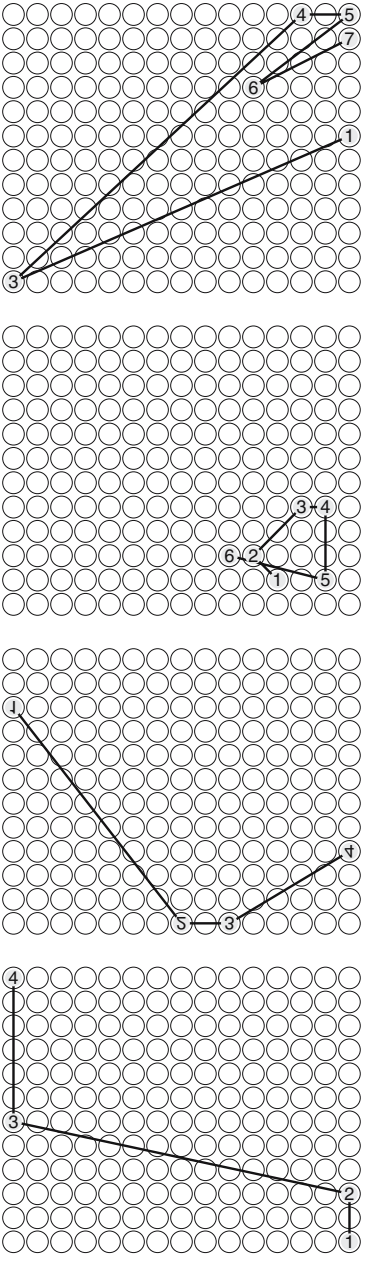


Figure 9.2. Trajectories along the river continua (1: sample site closest to the source); Ku (a), RW (b), ZO (c), References (d).

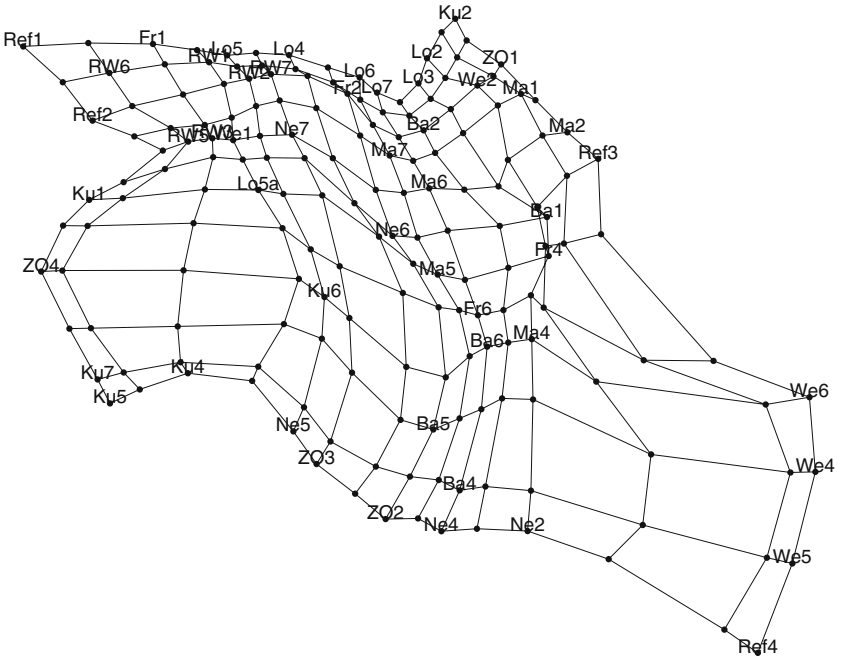


Figure 9.3. Similarities of the 55 samples with regard to eight water properties visualised by a Sammon map of a 15×12 SOM.

9.4 Correlations and Predictions with Supervised Learning Systems

Ecological systems (ecosystems) consist of the biological community and the abiotic environment. Many simultaneous and complex interrelations exist among the environmental variables, between the environment and the community, and within the members of the community. Many dependencies can be described by multiple non-linear regressions, a typical task for supervised learning ANNs. However, these are not necessarily cause-effect relations. The multitude of relationships typically implies redundant information. Important predictors either have to be selected or information must be compressed.

This is the required information to model relationships within the environment or the community, and to predict at least parts of the biological community from information on the environment and vice versa (bio-indication).

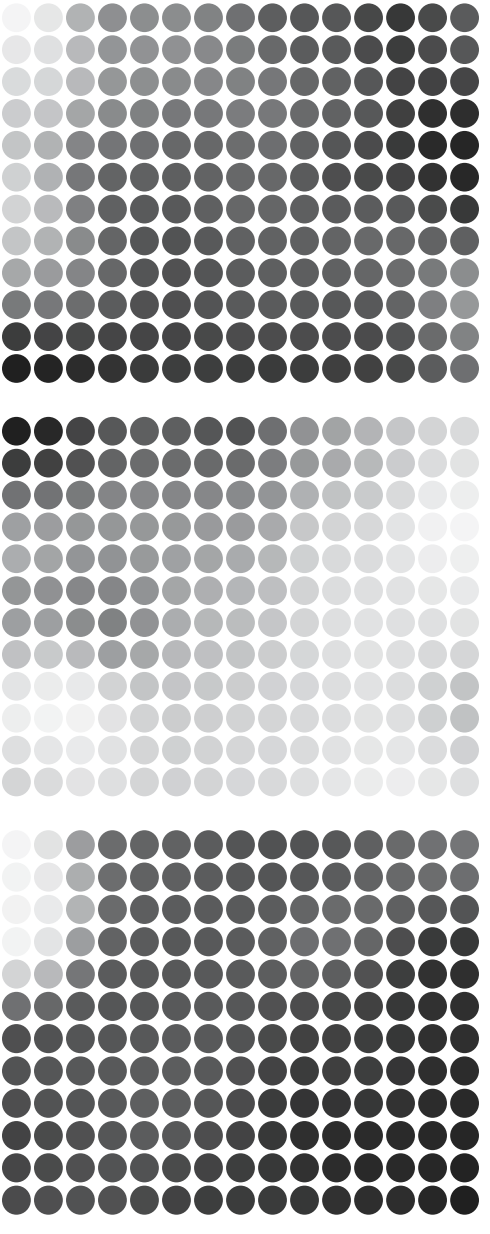


Figure 9.4. Planes of the individual water variables (light grey shade: high, dark grey shade: low values; see U-matrix-display in Fig. 9.2 containing the corresponding sampling sites); max. water temperature (a), min. oxygen (b), number of species per sampling site (c).

9.4.1
Correlations and Predictions of Environmental Variables

Data of physical and chemical water variables suffer from missing or wrong values due to instrument failures. It is possible to model costly or difficult to measure variables from cheaper and more precisely measured factors. Applications are error detection and correction of data sets as well as to fill gaps in data bases with more reliable values.

LNNs and a relatively small MLP network with only five neurons in one hidden layer were applied to model individual or groups of variables (e.g. conductivity, pH-value, O_2, NH_4-N, and NO_3-N). The resulting accuracy was high. Networks with only one output neuron had improved generalisation performance only for NH_4-N. This agrees with the narrow correlation of the variables conductivity, oxygen and pH among themselves and with the low correlation of these variables with the nitrogen variables.

All water quality variables regarded here were predicted with good accuracy ($B>0.7$) by a reduced number of network input variables (Borchardt et al 1997a). For dimension reduction the two methods 'regression' and 'sensitivity analysis' proved to be most suited (Dapper 1998).

9.4.2
Dependencies of Colonisation Patterns of Macro-Invertebrates on Water Quality and Habitat Characteristics

The search for interdependencies, coherence or even causality of environmental variables and a community is a fundamental challenge of ecology. The abundance of species and the community assemblage depend on the environment. An important aim in ecologically based research is the description of a species' environmental requirements facilitating the prediction of a community under given abiotic conditions.

9.4.2.1
Aquatic Insects in a Natural Stream, the Breitenbach

One goal was to predict the monthly species abundance of aquatic insects using a 'sliding time-window' on the data, first, of all environmental variables (discharge, precipitation, and water temperature) of the actual and the 12 preceding months, respectively and second, the species abundance of the 12 preceding months (Dapper 1998). This period comprises the one-year life-cycle of these insects. Second, the five most relevant predictors for ANNs were detected by a 'stepwise linear regression' (SPSS; Brosius 1995), and further, predictors were identified by neural sensitivity analysis (Dapper 1998). The resulting five-dimensional input vectors were computed with 20 MLPs. Finally, generalization ability of the best ANNs was visualized and compared to nets with all 51 input variables. The

resulting correlation coefficients of the models of 17 EPT species from the Breitenbach are compared in Table 9.1.

The determination coefficient B (on the 20% test data) was surprisingly high. It was >0.9 for five species, >0.7 for 9 species and >0.5 for 15 species. Only for *P. intricata* and *P. auberti* was B just below 0.5. The quality of the prognoses varied among species and among the different ANNs. This was either due to the variable abundances in the test data, or was related to the ecological plasticity of the populations. As expected, linear regression models had the lowest B values in almost all species. In *A. fimbriata*, *C. villosa*, *T. rostocki*, *P. auberti* and *P. intricata* linear regression and ANN models with pre-selection based on regression were of similar quality. Pre-selection of the five best variables by regression analysis (8 times) or by sensitivity analysis (6 times) improved prognoses. In three of seventeen species the reduction to the best five predictors was not accompanied by an increase of B. Only in the model for *L. nigra* did the use of all (51) predictors drastically increase B (by 0.29 or even 0.62) compared to models with the five most important variables. This may be due to the species life-cycle attributes (species traits). Larvae live on and in the relatively unstable sandy sediments and thus habitat and specimens are susceptible to almost every change in discharge throughout the year.

In summary, for most species models with dimension reduction by regression or sensitivity analysis produced models of similar quality (i.e. difference of $B \leq 0.05$). Pronounced differences were found for *L. prima* and *P. auberti* (pre-selection by sensitivity analysis), or *I. goertzi*, *P. intricata*, *S. torrentium* and *T. rostocki* (pre-selection by stepwise linear regression).

Reliability of all models was tested on the background of ANN computation and ecological knowledge. The results indicated that enhanced ecological flexibility of populations (risk spreading), low temporal resolution of the data, data scaling method, or different occurrences in learning and testing data resulted in a low model quality.

Scaling during pre-processing is one crucial step in exploring ecological data, and subsequent modelling. We transformed values linearly, sigmoidally, logarithmically and exponentially. Logarithmic scaling was optimal for discharge, to smooth extreme or rare events (floods). Prognoses for the abundance of *Baetis vernus* (and *B. rhodani*) after logarithmic scaling of all predictors resulted in a B=0.77 (compare Table 9.1 with linear scaled variables). However, after transforming the predicted values into original units, B was 0.63 or lower. Therefore, it appears that non-linear scaling did not improve any model.

Even though four different models with high determination coefficients were developed for four individual study sites (and six species), generalisation ability of the models was not as expected. Extrapolation on neighbouring sites at 600 m distance upstream or downstream was restricted.

Extreme values or low data density occurred in the different training data sets. If data points are dissimilar to the trained model, they should be attributed as novelty. To recognize those cases in general is part of the future work (novelty detection). The probability that many zero-predictions (i.e. no abundance in a particular month) had artificially increased the model quality and led to different

approaches (Obach et al. 2001). In addition, recurrent ANNs of the Jordan and Elman types were used without significant success.

Table 9.1. Overview of predictions of the abundance of aquatic insects in emergence traps with sliding windows technique; best B highlighted (all = all data of environmental variables and abundance of the parent generation included; regres. = 'most relevant' five predictors selected_by regression analysis; sensi. = 'most relevant' five predictors selected by sensitivity analysis; linear = linear regression. E = Ephemeroptera, P = Plecoptera, T = Trichoptera).

		best B predictors			
species	order	all	regress.	sensi.	linear
Baetis rhodani	E	0.56	**0.40**	**0.55**	0.40
Baetis vernus	E	**0.54**	**0.63**	0.71	0.54
Leuctra prima	P	**0.65**	**0.50**	0.74	0.44
Leuctra nigra	P	0.94	**0.65**	0.32	0.51
Leuctra digitata	P	**0.67**	0.91	0.89	0.48
Protonemura auberti	P	**0.32**	**0.38**	0.43	0.48
Protonemura intricata	P	**0.21**	**0.45**	0.30	0.48
Protonemura meyeri	P	**0.87**	0.91	0.91	0.34
Amphinemura standfussi	P	**0.86**	**0.91**	0.93	0.45
Isoperla goertzi	P	**0.54**	0.75	0.59	0.53
Siphonoperia torrentium	P	**0.34**	0.57	0.05	0.38
Apatania fimbriata	T	**0.63**	**0.86**	0.93	0.86
Chaetopteryx villosa	T	**0.35**	0.66	0.64	0.66
Drusus annulatus	T	**0.24**	**0.49**	0.56	0.08
Rhycophila fasciata	T	**0.58**	0.70	0.67	0.39
Tinodes rostocki	T	**0.45**	0.57	0.31	0.54
Agapetus fuscipes	T	0.56	**0.37**	0.44	0.34

9.4.2.2 Anthropogenically Altered Streams

Modelling macro-invertebrate communities depending on water quality or habitat characteristics was difficult. It was impossible to predict the entire set of 248 taxa as well as the ten best indicators for high organic load (DEV 1990) by eleven or seven water properties with MFM and MLP networks.

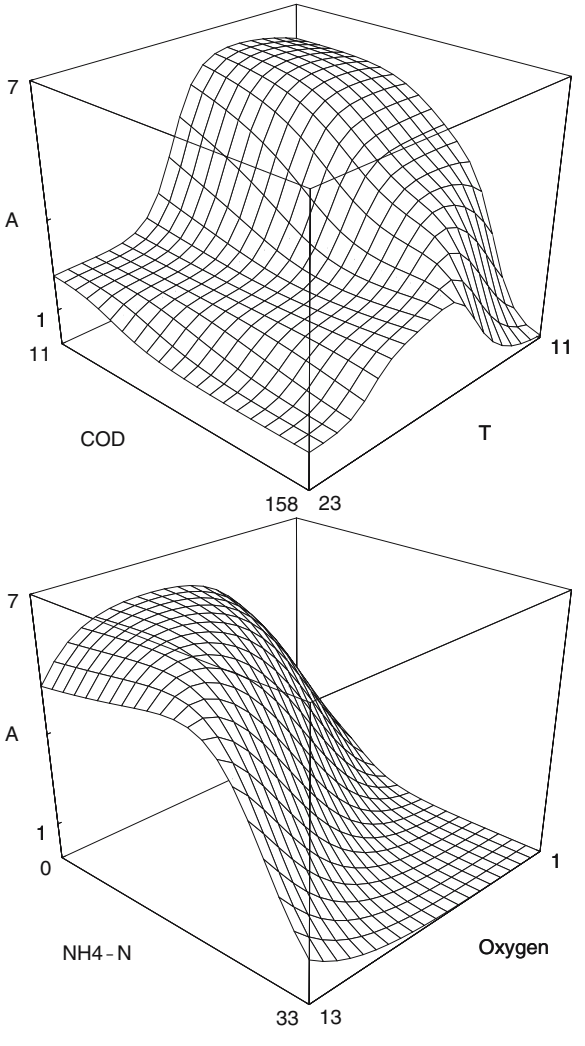


Figure 9.5. Prediction of *Gammarus pulex* abundance classes (A) based on 7 stream features with GRNN; two variables alter while the remaining 5 are kept constant at their mean values. T: maximum water temperature [°C]; oxygen, COD and NH_4-N in [mg/l].

The inappropriate ratio of cases and variables, the incompletely known ecological demands of species, and difficult analysis and training of networks with multiple output neurons led to models with single target variables (Borchardt et al. 1997a, Schleiter et al. 1999; 2001).

One of the most widespread organisms even in anthropogenically altered streams is *Gammarus pulex*. Abundance classes of this species were predicted with GRNN (i.e. kernel regression estimators) based on 14 stream features. Genetic algorithms selected the 7 most relevant variables: minimum O_2, the maximum of water temperature, COD, and NH_4-N, as well as channel characteristics like stream bed and bank structure, and longitudinal development of channel morphology. Importance (fitness) of predictors was calculated with a leave-one-out cross validation, suitable for small data sets.

Ecological model reliability is high because the selected variables are in good concordance with empirical knowledge of the species' distribution and related water quality measures (i.e. high abundance at intermediate water temperatures, high oxygen availability, and low pollution). Each figure (Fig. 9.5.) provides information on the variability of two predictors, the remaining were kept constant at their mean values (Obach 1998).

9.4.3 Bioindication

In the section above, we predicted community assemblage from a set of environmental variables. Here, we model the magnitude of abiotic characteristics, based on presence-absence and, in addition, abundance class data of indicator organisms. This is a basic task of bioindication. Because the variables may change in short time scales, biological indicators are adequate long-term probes for environmental quality. These organisms require and thus represent a defined environmental quality.

Initial experiments with species selected by MLPs from the anthropogenically altered stream data modelled, e.g. conductivity with a high precision based on abundance classes of only 40, 20, 10 or even 5 benthic macro-invertebrates (Schleiter et al. 1999; 2001). The reduction from initially 248 to only 5 taxa decreased the amount of input data by about 98% (Borchardt et al. 1997b).

Furthermore, various methods for predictor detection (e.g. sensitivity analysis on MLPs, linear regression analysis, genetic algorithms) were tested and the generalisation performance of different network types (MLPs with an additional special input layer, MFM, GRNN, LNN) was evaluated.

Eleven physical and chemical water quality measures, seven hydro-morphological habitat characteristics, and three combined quality indices (chemical and morphological water quality class, saprobic index) were modelled using presence-absence and abundance data of 127 (out of 248) species present on at least 10% of the 46 sites.

Conductivity was excellently modelled with both, 127 most frequent and presence of 10 species (RMSE=2.8 and 4.5% of range respectively). Other

chemical variables were predicted with high accuracy, with the exception of NH_4-N and N_{tot} (RMSE=14.5% of range). Data sets of a reduced number of species improved model quality for e.g. NO_3-N, pH, O_2, BOD_5. These species can be successfully used for bioindication.

For the majority of variables, presence-absence data provided better models compared with abundance classes. Predictability of hydro-morphological variables was sufficient only for the discharge regime and substrate diversity. Morphological quality class (Hessisches Ministerium für Umwelt, Landwirtschaft und Forsten 2000) was not predictable by macro-zoobenthos with the required accuracy. Saprobic index was estimated well based on only ten out of 60 indicator species (RescalRMSE=0.1).

Prediction of chemical water quality class (LAWA 1998) was easy and efficient. A linear regression model with presence-absence data of three predictors was sufficient. However, this prediction was not accurate on all test sites, because there are many reasons for a species' absence beyond the water quality. Only when other variables were verified by expert knowledge (e.g. intact stream morphology, community with several species), was the model reliable (Schleiter et al. 2001).

Depending on the network type and the selection method, different species groups were chosen for most, even for correlated output variables, and verified the results of Schleiter et al. (1999). Some species were useful for the prediction of several targets (e.g. *Gammarus pulex*: conductivity, saprobic index, oxygen, P_{tot}, NO_3-N; *Chironomus thummi*: conductivity, chemical water quality class, BOD_5, NO_3-N), whereas others appeared in only one model. Models with species groups selected here may not be generalized due to restrictions in the basic data set (narrow geographical region and limited abiotic gradients). Probably, other species not occuring in the DIN table (Friedrich 1990) may be suitable bioindicators for the saprobity in small streams. The results obtained need further validation based on additional data and expert knowledge.

The selection of relevant variables and the use of presence-absence data provided less complex, easier to understand and handle models and a drastic reduction of computational effort. This allowed an increased number of repetitions to provide more relevant results for generalisation. However, one problem is small training and test data sets. A low generalisation error may be accidental.

9.5 Assessment of Model Quality and Visualisation Possibilities: Hybrid Networks

A disadvantage of most ANNs are the complicated, difficult to comprehend internal network processes so that in many cases the neural networks are considered as black boxes.

Usually, the quality of neural network outputs is measured based on the difference between observed and predicted values. For quality assessments of the

network outputs, the error value alone is not sufficient because it does not provide any information concerning the reason of the error. Large differences can have their origin either in an unreliable model or in biased, unrepresentative data. Similarly, a small error may indicate a good model or, regarding outliers, a good answer just by chance. Therefore, the error measure can only help during the training phase. When the network deals with new input data, no output observations are available and hence the error can only be estimated on the basis of the local variability of output values corresponding to training inputs close to the actual input. Furthermore, the local reliability of the model depends on the amount of training data similar to the actual input vector and the distance to the nearest training vector.

We extended the combination of SOMs and RBF Networks to estimate the reliability of the network outputs and to get a better insight into the internal network activities. SOM-training optimised the centres of radial basis functions of an RBF Neural Network (Bishop 1995) and provided visualisation of the RBF-layer's activation patterns.

In analogy to the above studies, we predicted oxygen concentration based on macro-invertebrate species abundance (Werner and Obach 2001). Six test sites of 51 were randomly chosen. Fourteen predictors were selected by GRNNs and a genetic algorithm, which overcame the stepwise method.

The activations of the RBF neurons can be displayed for any input pattern on the SOM, which has been used previously for optimising the centres of radial basis functions. Thus it is possible to visualize and decide whether a test site is well known to the network or whether an extrapolation must be assumed (Werner and Obach 2001).

The variability of the output variable in classes determined by the SOM, can be displayed by box-and-whisker plots (Obach et al. 2001).

9.6 Conclusions

Artificial Neural Networks proved to be suitable tools to model non-linear interrelations in basic and applied ecology of running waters. Although they are able to describe correlations between multi-dimensional variables, causality detection is not possible. The major goal of model building is generalisation. Data samples must comprise as many information as possible and be representative for training and testing ANNs. For small data sets the performance of fast learning networks (LNN, GRNN, RBF) was estimated by cross-validation. Large amounts of data require pre-processing. Dimension reduction was performed with correlation analysis, multiple linear regression, principal component analysis, bottle-neck networks, sensitivity analysis, genetic algorithms and stepwise methods, depending on the applied ANN type and the linearity of the described correlation. Linear networks are simple in use, training and interpretation. They provide a benchmark against which the quality of other models like MLPs,

GRNNs, RBFNs, MFMs is compared. The application of different network types on the same problem is recommended.

We focussed on single network outputs because models with multiple outputs were more difficult to train and interpret. The quality of the models was described by the comparison of an ANN model error with trivial (persistence, naïve prediction) or easy to calculate (linear model, long-term mean) prediction errors. Our preferred error measure was the RMSE of [0,1]-scaled data, the ratio of the expected mean error from the range of the output variable's values. However, the RMSE in original units also provides valuable information to ecologists. These global errors do not necessarily estimate local reliability, which depends on e.g. the local variability of the output variable and data density. The combination of SOM and RBF networks (RBFSOM) combines good prediction properties on well supported input data with a warning function, if the particular input is not supported by training data, and hence the output information may not be valid. U-Matrix, Sammon map, visualization of input vector component planes and the display of neuron activities on the SOM codebook vectors are some graphical representation possibilities of the unsupervised trained SOMs. Feedforward network outputs can be displayed as 3D surfaces. The example of RBFSOM shows that it is profitable to connect different ANNs to a hybrid network in order to combine their special capabilities. Combined with GRNN for input relevance detection RBFSOMs become capable, efficient and transparent prediction tools.

We applied ANNs on data sets with environmental variables and communities to model interrelations in pristine and anthropogenically altered streams. The results confirmed interrelations between colonisation patterns of benthic macro-invertebrates, chemical and hydro-morphological habitat characteristics in lotic ecosystems. In a pristine stream discharge predominantly determined species abundances and community assemblage. Water temperature and other variables had smaller effects. High determination coefficients on test data were surprising, because of large proportions of trivial zero predictions. This inspired the application of more adequate tools and error measures.

On anthopogenic altered streams similarities of sampling sites as well as of individual variables were visualized with SOMs. Extraordinary sites were detected with Sammon maps and U-Matrix displays. Component planes were useful to analyze the responsible factors and to designate correlations among variables. Furthermore, leaps in the trajectories of individual streams on a map indicated abrupt changes of water quality.

Dependencies of species on their environment were modelled with ANNs circumscribing ecological niches. The effects of two important abiotic factors on the abundance classes of *Gammarus pulex* were displayed in 3D surface figures.

A reversed task is the prediction of environmental features from communities. Even information on the presence instead of the abundance of selected subsets of macro-invertebrates was adequate to perform bioindication of e.g. conductivity, oxygen and water quality classes with sufficient accuracy.

However, the last step in data analysis is the interpretation and the check of plausibility and the interpretation based on expert knowledge. ANNs have been used for more than a decade in ecology, but there are still many research fields

left. Further generalisation of the models beyond the area of Bunter Sandstone in Central Germany is the main future task beside the quantitative and qualitative extension of the data base.

Acknowledgements

The authors gratefully acknowledge the support by the German Research Foundation (Deutsche Forschungsgemeinschaft), grants No. We 959/5-1, We 959/5-2, BO 1012/5-3. We thank Dr Thomas Dapper, Klaus-Dieter Schmidt and numerous people who contributed e.g. in data collection.

References

Bayerisches Landesamt für Wasserwirtschaft (Ed.) (1998) Integrierte ökologische Gewässerbewertung – Inhalte und Möglichkeiten. Münchner Beiträge zur Abwasser-, Fischerei- und Flußbiologie 51, pp. 683

Bishop C (1995) Neural Networks for Pattern Recognition. Oxford University Press, Oxford

Borchardt D, Dapper T, Schleiter IM, Schmidt KD, Werner H, Wagner R (1997a) Modellierung von Wirkungszusammenhängen in Fliessgewässern mit Hilfe Neuronaler Netzwerke. Research report We 959/5-2, given to the German Research Foundation (Deutsche Forschungsgemeinschaft), Bonn, pp. 141

Borchardt D, Schleiter IM, Werner H, Dapper T, Schmidt KD (1997b) Modellierung ökologischer Zusammenhänge in Fliessgewässern mit Neuronalen Netzwerken. Wasser und Boden, 49(8), 38, 47--50

Brosius G, Brosius F (1995) SPSS Base System and Professional Statistics. International Thomson Publishing, Bonn

Chon TS, Park YS, Moon KH, Cha EY (1996) Patternizing communities by using an artificial neural network. Ecol. Modell., 90, 69--78

Dapper T (1998) Dimensionsreduzierende Vorverarbeitungen für Neuronale Netze mit Anwendungen in der Gewässerökologie. Dissertation im Fachbereich Mathematik/Informatik der Universität Gh Kassel. Berichte aus der Informatik D 34, Shaker Verlag, Aachen, pp. 234

DEV 1991. DIN 38410 Teil 2. Deutsche Einheitsverfahren zur Wasser-, Abwasser- und Schlammuntersuchung; Biologisch-ökologische Gewässeruntersuchung - Gruppe M.: Bestimmung des Saprobienindex M2. Beuth-Verlag, Berlin

Foody GM (1999) Applications of the self-organising feature map neural network in community data analysis. Ecol. Modell. 120, 97--107

Goldberg DE (1989) Genetic algorithms in search optimization and machine learning. 1st ed., Addison-Wesley, Reading MA, pp. 412

Hessisches Ministerium für Umwelt, Landwirtschaft und Forsten (Eds.) (2000) Erläuterungsbericht Gewässerstrukturgüte in Hessen 1999. Wiesbaden, pp. 25 and appendix

Kohonen T (1995) Self-Organizing Maps. Springer, Heidelberg

Länderarbeitsgemeinschaft Wasser LAWA (Eds.) (1998) Beurteilung der Wasserbeschaffenheit von Fließgewässern in der Bundesrepublik Deutschland – Chemische Gewässergüteklassifikation. 1st ed., Kulturbuchverlag, Berlin, pp. 35 and appendix

Obach M (1998) Anwendung Statistischer Methoden und Künstlicher Neuronaler Netzwerke im Vergleich. Diplomarbeit im Fachbereich Mathematik/Informatik der Universität Gesamthochschule Kassel (unpubl.)

Obach M, Wagner R, Werner H, Schmidt HH (2001) Modelling population dynamics of aquatic insects with Artificial Neural Networks. Ecol. Modell. 146, 1-3, 207-217.

Resh VH, Hildrew AG, Statzner B, Townsend CR (1994) Theoretical habitat templets, species traits and species richness: a synthesis of long-term ecological research on the Upper Rhône River in the context of concurrently developed ecological theory. Freshw. Biol. 31, 539--554

Ritter H, Martinetz T, Schulten K (1992) Neuronale Netze (2. ed.), Addison-Wesley, Bonn

Rumelhart DE, Hinton GE, Williams RJ (1986) Learning internal representations by error propagation. In: Rumelhart, D.E., McClelland, J.L.: Parallel Distributed Processing (Vol. I), MIT Press, pp. 318ff

Sammon JW (1969) A nonlinear mapping for data structure analysis. IEEE Trans. Comput. C-185, 401--409

Schleiter IM, Borchardt D, Wagner R, Dapper T, Schmidt KD, Schmidt HH, Werner H (1999) Modelling water quality, bioindication and population dynamics in lotic ecosystems using neural networks. Ecol. Modell. 120, 271--286

Schleiter IM, Obach M, Borchardt D, Werner H (2001) Prediction of running water properties using Radial Basis Function Self-Organising Maps combined with input relevance detection (i.prep.)

Schleiter IM, Obach M, Borchardt D, Werner H (2001) Bioindication of chemical and hydro-morphological habitat characteristics with benthic macro-invertebrates based on artificial neural networks, Aquat. Ecol. 35, 147--158

Specht DF (1991) A general regression network. In: IEEE Transactions on Neural Networks 6, 568--576

Statzner B, Resh VH, Dolédec S (Eds.) (1994) Ecology of the Upper Rhône River: a test of habitat templet theories. Special issue. Freshw. Biol. 31 (3) pp. 556

Townsend CR (1989) The patch dynamics concept of stream community ecology. J. N. Am. Benthol. Soc. 8, 36--50

Townsend CR, Hildrew AG (1994) Species traits in relation to a habitat templet for river systems. Freshw. Biol. 31, 265--275

Ultsch A (1993) Self organized feature maps for monitoring and knowledge acquisition of a chemical process. In: Gielen, S., Kappen, B. (Eds.), Proceedings of the International Conference on Artificial Neural Networks ICANN 93, 864–867, Springer, London

Vannote RL, Minshall GW, Cummins KW, Sedell JR, Cushing CE (1980) The river continuum concept. Can. J. Fish. Aquat. Sci. 37, 130--137

Wagner R, Dapper T, Schmidt HH (2000): The influence of environmental variables on the abundance of aquatic insects: a comparison of ordination and artificial neural networks. Hydrobiologia 422/423, 143--152

Werner H, Obach M (2001) New neural network types estimating the accuracy of response for ecological modelling. Ecol. Modell. 146, 1-3, 289-298.

Chapter 10

Non-linear Approach to Grouping, Dynamics and Organizational Informatics of Benthic Macroinvertebrate Communities in Streams by Artificial Neural Networks

T.-S. Chon · Y.S. Park · I.-S. Kwak · E.Y. Cha

10.1 Introduction

Benthic Macroinvertebrates in Streams

The topic of conservation of aquatic ecosystems and maintenance of water quality has been one of the utmost concerns as of late. The value of water, as for resources of drinking, agriculture, industry, energy and recreation, has been increasing rapidly due to the problems of water shortages and pollutions. Especially stream ecosystems flow through stressful sources, and are exposed to various natural and anthropogenic disturbances. Due to unique characteristics of streams such as continuous, one-way directional flow and complex relationships with the watershed area, streams convey various problematic agents rapidly, widely, and in a systematic way (Hynes, 1960; Calow and Petts 1994; Allan 1995; Hauer and Lamberti 1996; Welch and Lindell 1992).

One of the key issues in maintaining aquatic ecosystems is to find efficient and sustainable management strategies of biological communities residing in streams. Biological communities equally share "the water right" as humans do, which, however, has been seriously neglected or ill considered in numerous land-use plannings through the period of rapid industrial development in the 20th century. Biological communities have close interrelationships with their habitats: their residence is based on the habitat suitability, while communities in turn contribute to determine characteristics of habitats in the context of the Gaia hypothesis. Biological communities reveal ecological functions of stream ecosystems, and are direct indicators of ecosystem health.

Among various biological communities found in streams, benthic macroinvertebrates have been considered as one of the most important taxa. They play a key role in food web dynamics, linking producers and top carnivores, and are one of the most reliable indicator groups along with algae in freshwater ecosystems (Hellawell 1986). Their spatial sedentariness and intermediate life span -from several months to several years- make macroinvertebrates ideal as for

an integrative and continuous indicator group of water quality (Hynes 1960; Hawkes 1979; Sladecek 1979; Tittizer and Kothe 1979; Hellawell 1986). Many useful biological indicators based on benthic macroinvertebrates such as TBI, BMWP have been developed (e.g., Spellerberg 1991).

Benthic macroinvertebrates are generally cosmopolitan and diverse. The parameters on community structure such as diversity and dominance could be effectively used for indicating water quality as well as for expressing ecological status. At the same time, each different group of macroinvertebrates could be the indicator to specific toxic effects. The physicochemical analyses are specific and accurate, however they are sometimes not integrative, would provide only local information, and are generally expensive. Monitoring by biological communities could be a good compensator for the physicochemical methods for indicating water quality (Hellawell 1986; Tittizer and Kothe 1979; Welch and Lindell 1992).

Artificial Neural Networks and Non-Linear Data

Through field survey, data for community dynamics are usually obtained from various sample sites on the regular basis, and are accumulated during the survey period (e.g., Fig. 8.1). Since communities consist of many species and vary in nonlinear fashions, they are complex and difficult to analyze. There have been numerous accounts of statistical analyses on communities through conventional multivariate analyses (e.g., Bunn et al. 1986; Legendre and Legendre 1987; Ludwig and Reynolds 1988; Quinn et al. 1991). The researches have been usually directed to classification of communities to ordination of multivariate data through eigen analyses. However, conventional statistical methods are mainly limited to linear data (Ludwig and Reynolds 1988), and are not flexible in many aspects, for instance, data handling (e.g., missing samples) and predicting dynamics.

Artificial neural networks solve this problem of complexity in community data. Artificial neural networks are parallel and distributed information extraction processors, have adaptive and self-organizing properties, and are consequently feasible in handling nonlinear data (Lippmann 1987; Hecht-Nielsen 1990; Zurada1992; Haykin 1994). Since the neural computation system was proposed by McCulloch and Pitts (1943) in the forties, artificial neural networks have been rapidly developed in extracting information of complex and nonlinear phenomena in a wide spectrum in the field of machine intelligence in the eighties (e.g., Lippmann 1987; Wasserman 1989; Hecht-Nielsen 1990; Zurada 1992; Haykin 1994)). In ecology, artificial neural networks have been used for classifying groups (e.g., Chon et al. 1996; Levine et al. 1996), and for patterning complex relationships (e.g., Lek et al. 1996; Huntingford and Cox 1996; Tuma et al. 1996). On macro-invertebrates in aquatic ecosystems, training with artificial neural networks have been conducted on grouping and community dynamics (Chon et al. 1996, 2000a, 2000b, 2000c, 2001; Park et al. 2001a, 2001b; Brosse et al. 2001). Implementations of artificial neural networks to ecology have been extensively reviewed by Lek and Guegan (1999, 2000).

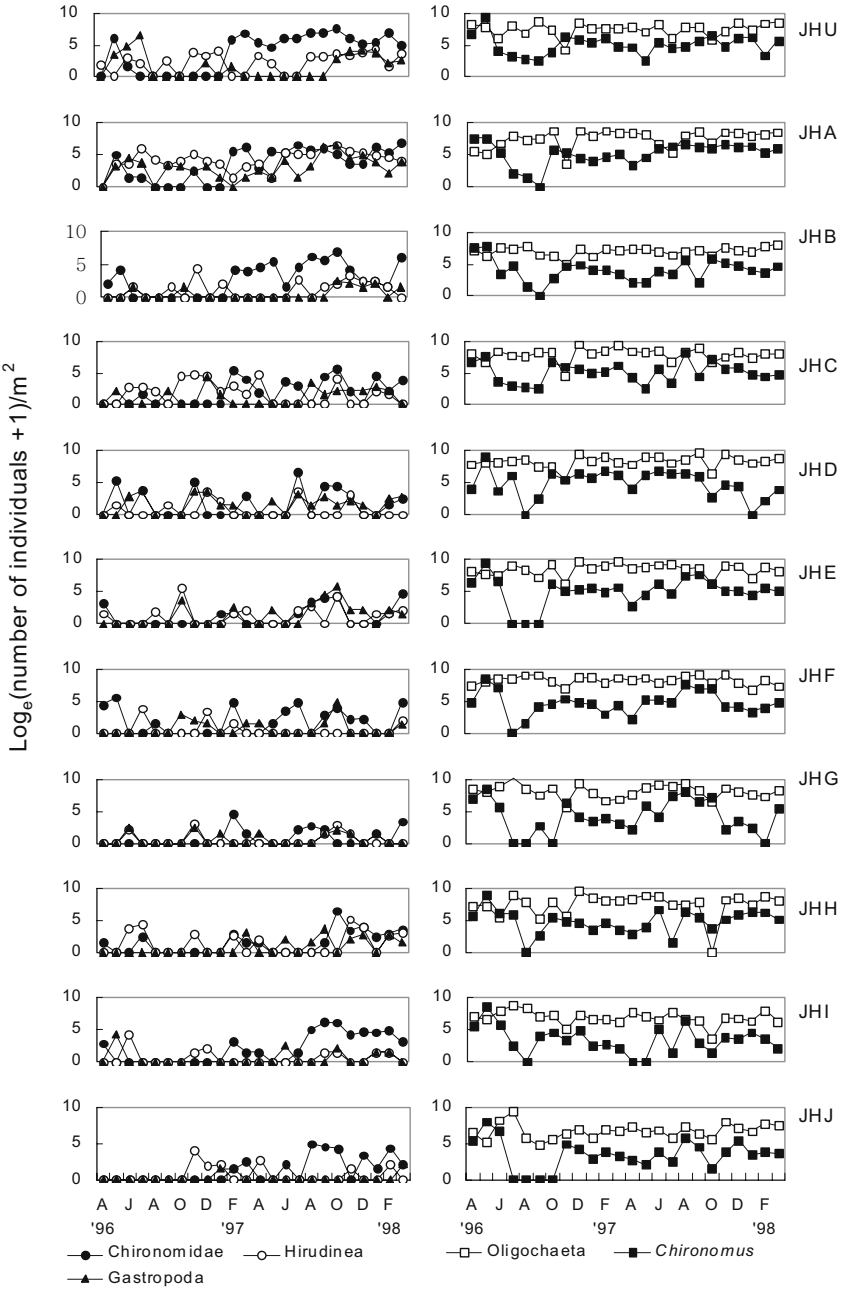


Fig. 10.1. Monthly changes in densities (log-transformed) of selected taxa of benthic macroinvertebrate community at the sampling sites in the Yangjae Stream, Han River, Korea, from April 1996 to March 1998. (From Chon et al. 2001).

In this chapter, based on our experiences on filed data, we try to demonstrate how artificial neural networks could be utilized as a general tool for analyzing and predicting macroinvertebrate communities in streams. We try to elucidate feasibility of artificial neural networks in grouping of communities as classification and ordination, predicting multivariate community dynamics, verifying environmental impacts, and revealing organizational aspects of community.

10.2 Grouping Through Self-Organization

10.2.1 Static Grouping

Kohonen Network

Community data are complex and difficult to analyze as mentioned previously. After collection of samples, the first step required is to have a comprehensive view on the overall pattern of the collected community samples. This could be generally conducted by classification or ordination through conventional statistical methods. Through clustering the communities were grouped in a hierarchical manner dependent upon the degree of similarity among the sampled communities (Ludwig and Reynolds 1988). Based on eigen analysis approach, associations among sample communities (e.g., Q mode) or variables such as taxa and environmental factors (e.g., R mode) could be revealed on principal factors through ordination (Legendre and Legendre 1987; Ludwig and Reynolds 1988).

As mentioned previously, however, the conventional methods are generally limited to linear data. Artificial neural networks is an alternative tool for community classification, and the self-organizing mapping (SOM) is useful for grouping non-linear data. The Kohonen network (Kohonen 1989) is one of the most frequently used models for self-organizing, and the network has been successfully implemented to patterning community data (e.g., Chon et al. 1996; Foody 1999; Giraudel et al. 2000). The Kohonen network extracts information out of multi-dimension data and maps onto the space of the reduced dimension (e.g., 2 or 3). In the Kohonen network, in this study, a linear array of M^2 artificial neurons (i.e., computation nodes), with each neuron being represented as j (Fig. 10.2) is arranged in two dimensions for the convenience of visual understanding (Chon et al. 1996). Suppose a community data containing N species (i.e., N dimensions), and the density of species, i, is expressed as a vector x_i. The vector x_i is considered to be an input layer to the Kohonen network. In the network each neuron, j, is supposed to be connected to each node, i, of the input layer. The connectivities are represented as weights, $w_{ij}(t)$, adaptively changing at each iteration of calculations, t. Initially the weights are randomly assigned in small

values. When the input vector is sent through the network, each neuron of the network computes the summed distance between the weight and input as shown below:

$$d_j(t) = \sum_{i=0}^{N-1} (x_i - w_{ij}(t))^2 \tag{10.1}$$

The neuron responding maximally to a given input vector is chosen to be the winning neuron, the weight vector of which has

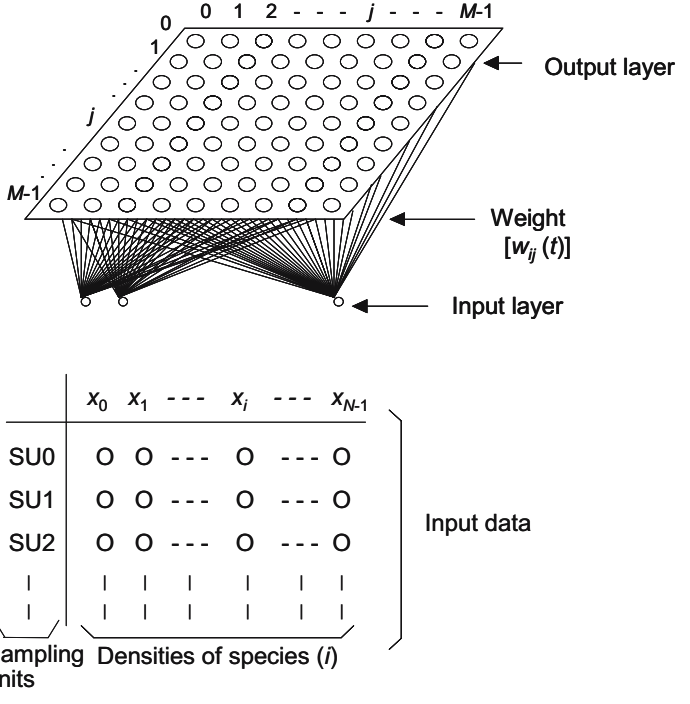

0 1 2 - - - j - - - M-1
0
1
j
M-1
Output layer
Weight
[w_{ij} (t)]
Input layer
x_0 x_1 - - - x_i - - - x_{N-1}
SU0 O O - - - O - - - O
SU1 O O - - - O - - - O
SU2 O O - - - O - - - O
Input data
Sampling units
Densities of species (i)

Fig. 10.2. Schematic diagram of the Kohonen network. (From Chon et al. 1996).

the shortest distance to the input vector. The winning neuron and possibly its neighboring neurons are allowed to learn by changing the weights in the manner to further reduce the distance between the weight and the input vector as shown below:

$$w_{ij}(t+1) = w_{ij}(t) + \eta(t)(x_i - w_{ij}(t))Z_j \tag{10.2}$$

where Z_j is assigned 1 for the winning (and its neighboring) neuron(s) while it is assigned 0 for the rest neurons, and $\eta(t)$ (e.g., 0.1 - 0.4) denotes the fractional increment of the correction. The radius defining neighborhood is usually set to a larger value early in the training process, and is gradually reduced as convergence is reached. Detailed algorithm could be referred to Kohonen (1989), Hecht-

	0	1	2	3	4	5	6	7	8
0			ST3SU ST4A ST4W		ST2SU ST4SP ST4SU		ST1A ST2A ST3A		
1	ST3SP								SY4W
2					ST1SU ST2SP		SY2A SY5A		
3	ST2W ST3W								SY1A SY3A SY4A
4				SY1SP		SY2SP SY3SP SY4SP SY5SP			
5	CM2W SY2W SY3W SY5W								CM2SU CM3SU CM4SU CM5SU
6			CM5A		CM1SU SY2SU		SY5SU		
7									
8	SY1W ST1W ST1SP		CM1SP CM2A SY1SU SY3SU		CM1A CM2SP CM3A CM3SP CM4A CM5SP		CM1W CM3W CM4W CM4SP CM5W		SY4SU

Fig. 10.3. Mapping of the benthic macroinvertebrate communities collected at the study sites in the Suyong River by the Kohonen network after training. (The first two characters and the next one numeric digit stand for study sites, where CM, ST and SY represent the Cholma, Soktae and Suyong streams of the Suyong River respectively. The last one or two characters appearing at the end of study sites represent seasons when the samples were collected: SP; spring, SU; summer, A; Autumn, and W; winter.) (From Chon et al. 1996).

Nielsen (1990), Zurada (1992) and Chon et al. (1996). The grouping was conducted on benthic macro-invertebrate communities collected at the sample sites in urbanized streams in the Suyong River in Korea in different seasons. The total number of species (i.e., number of nodes in the input layer) used for training was 99 and the number of collected samples for input data was 60. The general ecological assessment on the Suyong River has been reported in Kwon and Chon (1993). The input values with greatly different numerical values in densities are avoided for training. In this case the data were transformed by natural logarithm in order to emphasize differences in low densities, and, subsequently, the transformed data were proportionally normalized between 0.01 and 0.99 in the range of the maximum and minimum density for each species collected during the survey period.

Fig. 10.3 shows an example of grouping by the Kohonen network with a mapping of 9 × 9 neurons (Chon et al. 1996). The convergence was mostly reached in 500 - 1000 iterations. Communities were grouped according to different impacts of pollution and topography of the sample sites. The area of the map was divided according to the main tributaries (ST, CM and SY), and grouping was influenced by impacts of pollution.

Classification could be also conducted by the conventional clustering analysis (Ludwig and Reynolds, 1988). The same input given to the Kohonen network were provided to the clustering analysis utilizing the method of average linkage between groups (Norusis 1986). The clustering results were in general similar to those by the Kohonen network (Fig. 10.3), and confirmed overall groupings by the Kohonen network. Benthic macroinvertebrates responded differently in groups to anthropogenic impacts of pollution from oligo-saprobity to poly-saprobity as the stream flowed down. Communities collected at higher saprobities showed higher levels of similarities, suggesting higher degree of closeness obtained among the communities collected at polluted sites.

It was generally difficult to directly compare performance of groupings by the two methods, clustering and SOM. Since communities were grouped in an unsupervised manner, there are no objective references of groupings to be compared with. Based on experience with field data, however, mapping by the Kohonen network appeared to be more realistic. The same groupings were observed between SOM and the clustering in some case. For example, the group of CM2SU, CM3SU, CM4SU and CM5SU (neuron (8 (*x* axis), 5 (*y* axis)) and that of SY2W, SY3W, SY5W and CM2W (neuron (0,5)) in SOM in Fig. 10.3 correspondingly matched to the same sample groups on clustering (Fig. 10.4). In the other groups, however, discrepancies were observed. For example, the samples in the group of CM1SP, CM2A, SY1SU and SY2SU (neuron (2,8)) in the Kohonen mapping (Fig. 10.3) were all scattered in the clustering analysis (Fig. 10.4). The communities patterned at this neuron, however, were more similar to field data, and the grouping in the Kohonen network appeared to be more realistic. Groupings in other cases (e.g., "SY2A and SY5A (neuron (6,2))" and "ST1SU and ST2SP (neuron (4,2))") in Fig. 10.3 also tended to reflect more field situations than the groupings by the clustering analysis in the than groupings by the clustering analysis.

The overall conformation of groupings was also more explainable in the map of the Kohonen network. The neurons representing the three tributaries of the Suyong River were clearly divided in the Kohonen network (Fig. 10.3). The Soktae Stream (ST1 - ST4) occupied mainly a large triangular area at the upper left part of the map. The most neurons representing the Cholma Stream (CM1 - CM5) were located around the bottom right corner of the map, while those designating Suyong Stream (SY1 - SY5) generally occupied a diagonal area from the upper right to bottom left corner of the map. The communities patterned at the upper left corner were in general highly polluted, including some sample sites in the Soktae Stream. This polluted area was bordered with the area of the intermediate pollution, diagonally starting from neuron (0, 3) to the upper right corner of the

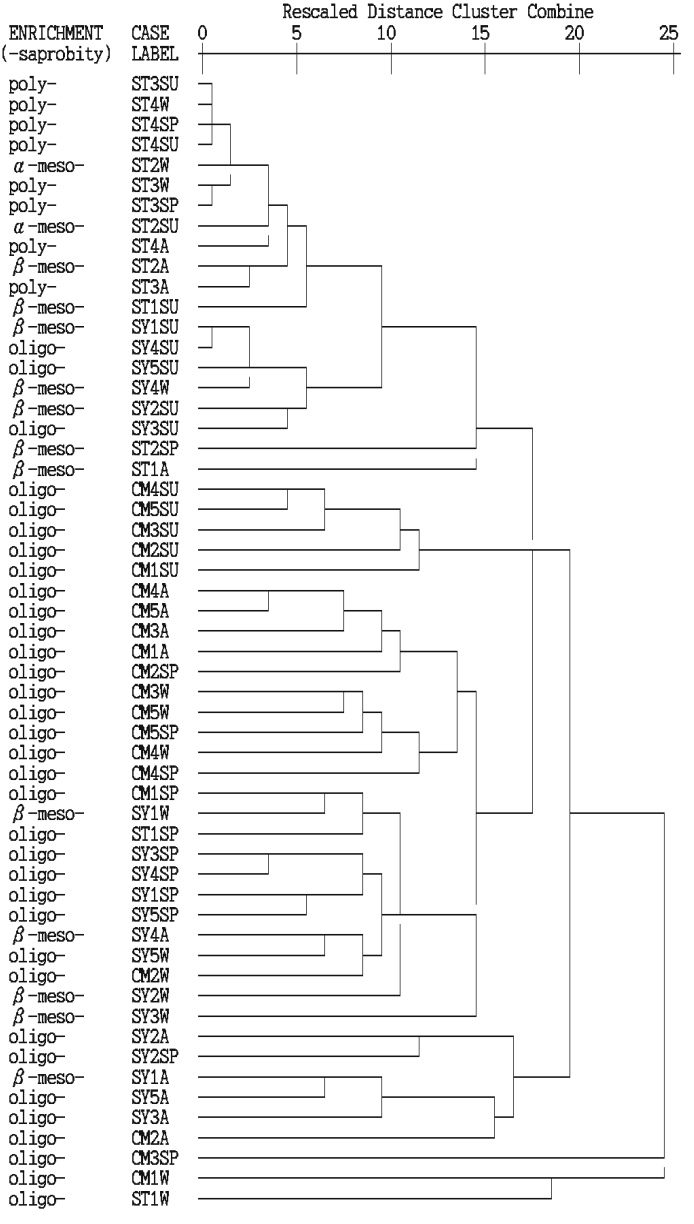


Fig. 10.4. Clustering benthic macroinvertebrate communities collected at the study sites in the Suyong River in 1989. (The name of communities are explained in the caption of Fig. 3. The overall saprobity level for each site appears under the first column of ENRICHMENT.) (From Chon et al. 1996).

map, which includes the sites of ST2W, ST3W, ST1SU, ST2SP, ST1A, ST2A, ST3A, SY4W, etc. The area further below the border zone was mainly occupied by the relatively clean sample sites of the Cholma Stream along with other sites in the Suyong and Soktae streams. Consequently, the mapping area appeared to be divided according to the impact of pollution and topography of the streams. In the clustering (Fig. 10.4), however, this organization of communities was not clearly observed. The sample sites were mainly lined up in different order of saprobic levels.

Within these broad topographical dispositions of communities in the map of the Kohonen network, the trained communities further appeared be organized in small scale. For example, the group of CM2SU, CM3SU, CM4SU, and CM5SU occurred in the same season (summer), while the other groups also appeared according to different seasons (Fig. 10.3). This indicated that sampled communities were organized in topographical dispositions firstly, and in seasons secondly. This suggested the possibility of hierarchical organization in data grouping in SOM.

The Kohonen network not only allows grouping but also makes it possible to *patternize* new data, by assigning a new component (i.e., neuron). When a newly collected community is given to the network as an input, it may be recognized either as one of the already-determined patterns or as a new pattern (Chon et al. 1996). The newly recognized results could be compared with the trained patterns (Fig. 10.5). At the polluted sites in this case, the new data for ST4A and ST4W,

	0	1	2	3	4	5	6	7	8
0			ST4A ○ ST4W ○ ST4A ● ST4W ● ST4SU ●		ST4SP ○ ST4SU ○				
1									
2									
3	ST4SP ●								
4									
5									
6									
7									
8									

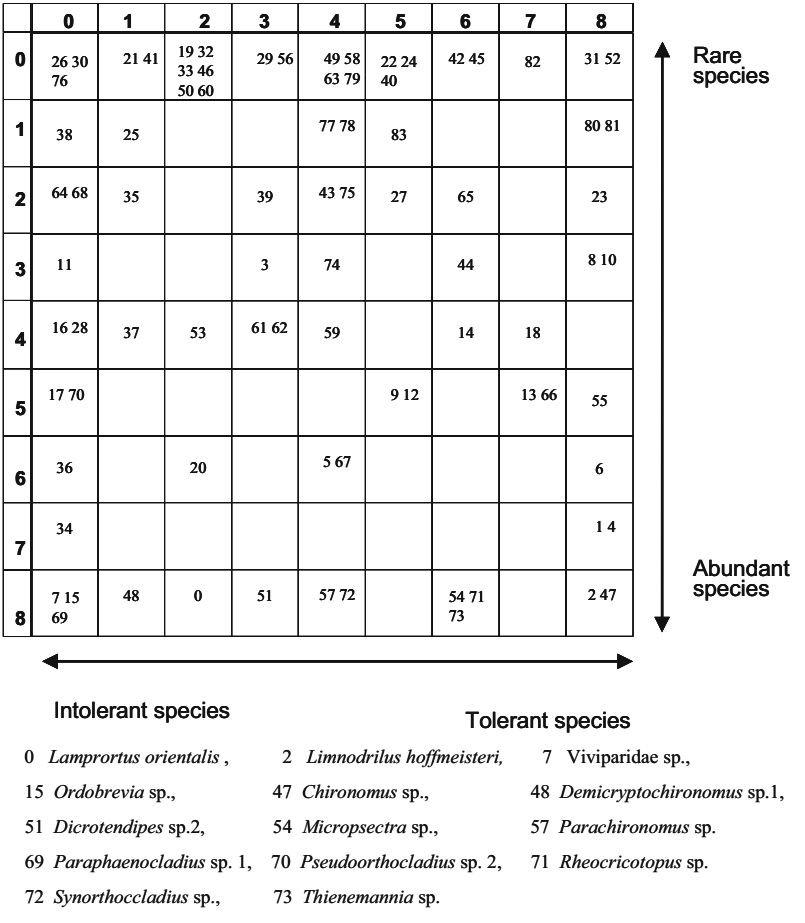
Fig. 10.5. Recognition of benthic macroinvertebrate communities collected at ST4 in the Soktae Stream in 1992 on the trained Kohonen map. (The name of communities are explained in the caption of Fig. 10.3. The black and white circles Although the Kohonen network appeared to be a classifier of communities in this case, the network actually could also serve as appearing at the end of the community respectively represent the recognized and trained patterns.) (From Chon et al. 1996).

for example, were matched to the trained patterns. This concept of patternizing may appear as a notable process in interpreting ecological data.

Although the Kohonen network appeared to be a classifier of communities in this case, the network actually could also serve as an ordination tool. From the aspect of reducing dimensions, the Kohonen network is basically similar to Principal Component Analysis (PCA): input data dimensions are effectively contracted to a limited number of dimensions in output (e.g., 2 or 3 dimensions). Not only for sample communities (Q mode), different taxa (R mode) also could be grouped on SOM. Fig. 10.6 shows mapping of the selected taxa in benthic macroinvertebrate communities collected at Cholma, Suyong, Heodong, and Seoktae streams in the Suyong River through training by the Kohonen network. The species were classified according to the gradient of pollution and abundance. The right area of the map was represented by pollution tolerant species, while the left area was occupied by pollution intolerant species. For example, *Limnodrilus hoffmeisteri* and *Chironomus* sp., which were collected in streams polluted by organic matters, were grouped in the lower right (neuron (8, 8) (No. 2)), while Viviparidae (No. 7), *Ordobrevia* sp. (No. 15), and *Paraphaenocladius* sp. (No. 69), which were collected at relatively unpolluted streams, were located at the lower left area of the map (neuron (0,8)). The figure also showed that the lower area was patterned with abundant species, while the upper area was occupied by relatively rare species. Ecological explanation on associations of different taxa of benthic macroinvertebrates in the Suyong River will be reported elsewhere.

Melssen et al. (1993) mentioned that the huge number of data variables may yield a larger number of significant principal components in PCA so that it may not retain sufficient information if only a few principal components are used for visualizing the multidimensional data space. Also some computational problems might arise due to the large number of variables, such as calculating (pseudo-) inverse of the covariance matrix. However the Kohonen network, trained in an unsupervised fashion, could be utilized to map the multidimensional data space on two or a few more dimensions, preserving the existing topology as much as possible. Lohninger and Stanc (1992) compared the Kohonen mapping and the k-nearest neighbour clustering in classification of mass spectral data in chemical compositions. They reported that the former was superior in all cases they tested. The comparison between the Kohonen network and statistical clustering methods is further discussed in Chon et al. (1996).

Although each neuron patternizes a group of similar communities and the neurons representing communities under similar environmental conditions are generally located in groups on the map, the distances among patternized neurons measured on the map may not directly indicate the degree of closeness among communities. Interpreting the distances among neurons on the trained map is a complicate problem, considering that the original multivariate data set was transformed into a space of a few dimensions. Further investigations are needed to express the degree of associations among communities in a more feasible manner in reduced dimensions on the map.


Rare species
Abundant species
Intolerant species
Tolerant species
0 Lamprortus orientalis , 2 Limnodrilus hoffmeisteri, 7 Viviparidae sp.,
15 Ordobrevia sp., 47 Chironomus sp., 48 Demicryptochironomus sp.1,
51 Dicrotendipes sp.2, 54 Micropsectra sp., 57 Parachironomus sp.
69 Paraphaenocladius sp. 1, 70 Pseudoorthocladius sp. 2, 71 Rheocricotopus sp.
72 Synorthoccladius sp., 73 Thienemannia sp.

	0	1	2	3	4	5	6	7	8
0	26 30 76	21 41	19 32 33 46 50 60	29 56	49 58 63 79	22 24 40	42 45	82	31 52
1	38	25			77 78	83			80 81
2	64 68	35		39	43 75	27	65		23
3	11			3	74		44		8 10
4	16 28	37	53	61 62	59		14	18	
5	17 70					9 12		13 66	55
6	36		20		5 67				6
7	34								1 4
8	7 15 69	48	0	51	57 72		54 71 73		2 47

Fig. 10.6. Mapping of the selected taxa of benthic macroinvertebrate communities collected in the Cholma, Suyong, Heodong, and Soktae streams in the Suyong River. (The names of taxa appearing only at the last two strips from the bottom of the map are listed on the figure for the purpose of simplicity of explanation.)

Adaptive Resonance Theory

Since artificial neural networks have adaptive and self-organizing properties, other models are also feasible in organizing community data (e.g., Kamgar-Parsi et al. 1990). One of the most frequently used networks for classification is the Adaptive Resonance Theory (ART; Carpenter and Grossberg 1987; Pao 1989). In ART (see Fig. 10.7b), bottom up weights, $b_{ji}(0)$, between output node j and input node i were initialized with some small numbers. After the input x_i , density and species richness in selected taxa, is given to the network, distance, $d_j(t)$, for each output node, j, is calculated as follows:

$$d_j(t) = \sqrt{\sum_{i=0}^{n-1} [b_{ji}(t) - x_i]^2} \tag{10.3}$$

where n is the number of input nodes. The distance, $d_j(t)$, measures the degree of similarity between weights and input data, and is used as a criterion for grouping inputs through the training process.

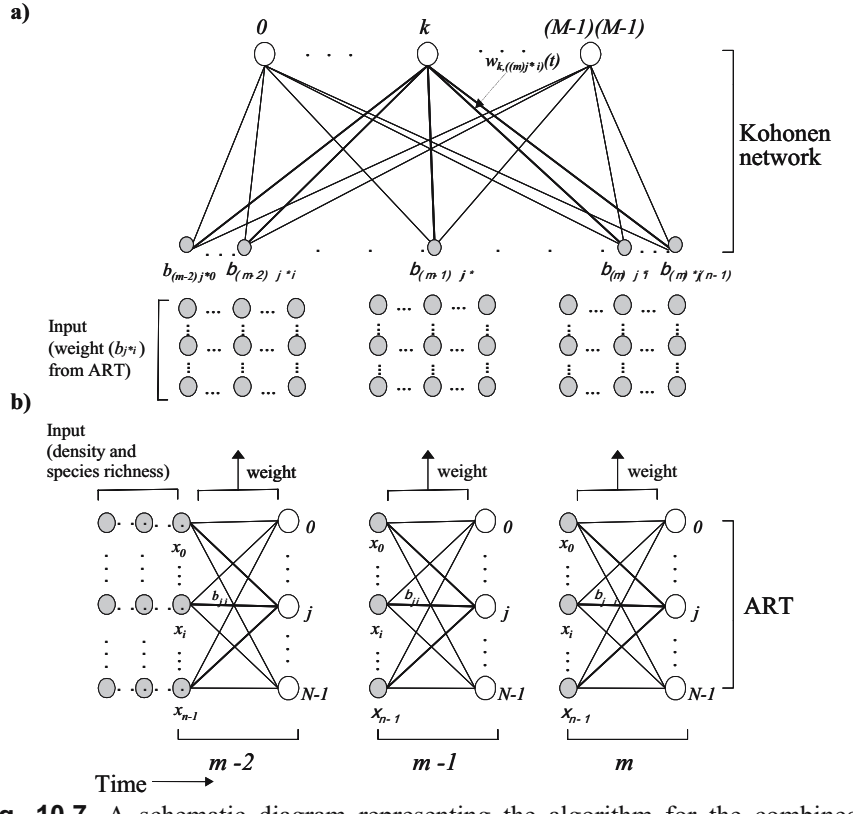


Fig. 10.7. A schematic diagram representing the algorithm for the combined network of ART (a) and Kohonen (b) for grouping community changes (m ; sampling month in the sequential period, n; number of input nodes for ART, N; number of output nodes for ART, x_i ; input data at the node i in ART, b_{ji} ; bottom up weights between output node j and input node i in ART, b_{j*i} ; converged weight of ART which is used as input data in Kohonen network, M; order of output node for the Kohonen network, $w_{k,(mj^*i)}(t)$; weight in Kohonen network). (From Chon et al. 2000c).

As each new input enters the network the distance is calculated and the output node j which has minimum distance is selected as j^*. If $d_{j^*}(t)$ is smaller than ρ, which is a threshold parameter in determining vigilance and determined based on efficiency of grouping of input data, the input is assigned to output node j^*. The weight for the node j^*, $b_{j^*i}(t)$, then, is updated as follows:

$$b_{j^* i}(t+1) = \frac{c}{c+1} b_{j^* i}(t) + \frac{1}{c+1} x_i \tag{10.4}$$

where c is the number of sample units classified to node j^*.

If $d_{j^*}(t)$ is larger than ρ the input is assigned to new output. This means that the entering input forms to a new pattern, not belonging to one of the previously existing patterns. Then its weight $b_{j^*i}(t)$ is newly assigned as follows:

$$b_{j^* i}(t+1) = x_i . \tag{10.5}$$

The benthic macroinvertebrate communities collected monthly in the Suyong and Soktae streams in the Suyong River, Korea, from September 1993 to August 1994, were used for input data. Since the number of species (132) were too many to train at PC (Personal Computer) level, species were grouped to 7 important taxa such as Chironomidae, Diptera (except Chironomidae), Trichoptera, Ephemeroptera, Miscellaneous Insecta, Oligochaeta, and Miscellaneous Macroinvertebrate. Densities (number of individuals per m^2) and species richness (number of species) in each important taxa, as well as the total density and the total species richness, were given as inputs for training with ART. The total number of input node for ART was sixteen.

These weights produced by ART preserve the conformational characteristic of input data for grouping, and through them the associations among the communities are projected into the space defined in ART (Zurada, 1992). Fig. 10.8a is an example of classification by ART (Chon et al. 2000c). The number of trained communities was 84, while the numbers of output nodes for ART was eleven. The threshold for vigilance, ρ, was set to 0.61. Similar to the case of groupings by the Kohonen network (Fig. 10.3), communities were generally classified according to topographical conditions and degree of pollution. Sample sites from the same stream had a higher tendency of grouping. Also sample communities collected from the polluted sites such as TCL and THP were grouped closely, and were separated from the other less-polluted sites. The community data collected from TKC, which were in a medium range between clean and polluted status, appeared to be diverse, and were divided into many small groups, separately (e.g., TKC4-2 and TKC4-3 of neuron 7 in Fig. 10.8a), or in groups with the other similar sites (e.g., TKC3-9 and TKC4-8 of neuron 0 in Fig. 10.8a).

a)

Neuron	Number	Sampling units
0	14	YIG3-9 YIG3-10 YIG3-11 YIG4-7 YCK3-9 YCK3-10 YCK4-3 YSC3-9 YSC3-10 YSC3-12 YSC4-7 TSD3-9 TKC3-9 TKC4-8
1	12	YIG3-12 YIG4-1 YIG4-2 YIG4-3 YIG4-4 YIG4-5 YCK3-12 YSC4-2 YSC4-4 TKC4-1 TKC4-4 TKC4-5
2	12	YIG4-6 YIG4-8 YCK4-1 YCK4-2 YCK4-4 YCK4-5 YCK4-6 YCK4-7 YCK4-8 TSD4-1 TSD4-4 TSD4-7
3	6	YCK3-11 YSC3-11 YSC4-6 YSC4-8 TSD4-5 TKC3-10
4	2	YSC4-1 YSC4-3
5	3	TSD3-10 TSD3-11 TSD3-12
6	4	TSD4-2 TSD4-3 TSD4-6 TSD4-8
7	2	TKC4-2 TKC4-3
8	2	TKC4-6 TKC4-7
9	8	TKC3-11 THP3-9 THP3-10 THP3-11 THP4-6 TCL3-11 TCL3-12 TCL4-6
10	19	YSC4-5 TKC3-12 THP3-12 THP4-1 THP4-2 THP4-3 THP4-4 THP4-5 THP4-7 THP4-8 TCL3-9 TCL3-10 TCL4-1 TCL4-2 TCL4-3 TCL4-4 TCL4-5 TCL4-7 TCL4-8

Fig. 10.8. Patterns of benthic macroinvertebrate communities in one-month samples, collected in the Suyong River as structured by ART (a), and the Kohonen network (b). In (a), sample communities associated with specified neurons through ART training are listed in groups. In (b), the neurons were arranged in two dimensions, and sample communities patterned by the Kohonen network to a specified neuron (*i*, *j*) are grouped together in the associated table position (*i*, *j*). (The alpha-codes, three characters in the figure, designates the name of sample sites: TSD; Sadeungkol, TKC; Kochon, THP; Hapansong, TCL; Chungli, YIG; Imgog, YCK; Changki, and YSC; Shinchon. The first numerical digit appearing after the alpha-codes represents the year of collection (i.e., 3 for 1993 and 4 for 1994) while the second numerical digit following the dash stands for the month of collection (e.g., 1 for January, 2 for February, etc.).) (From Chon et al. 2000c).

b)

	0	1	2	3	4	5	6	7
0						THP3-9 TKC3-10 TKC3-11 THP3-11 TCL3-11 TCL3-12 THP4-6 TCL4-6		
1								
2								
3	TCL3-9 TCL3-10 TKC3-12 THP3-12 THP4-1 TCL4-1 THP4-2 TCL4-2 THP4-3 TCL4-3 THP4-4 TCL4-4 YSC4-5 THP4-5 TCL4-5 THP4-7 TCL4-7 THP4-8 TCL4-8							YIG3-12 YCK3-12 YIG4-1 TKC4-1 YIG4-2 YSC4-2 YIG4-3 YIG4-4 YSC4-4 TKC4-4 YIG4-5 TKC4-5
4				TKC3-10 YCK3-11 YSC3-11 TSD4-5 YSC4-6 YSC4-8				
5								
6								
7	TSD3-10 TSD3-11 TSD3-12	TSD4-2 TSD4-3 TSD4-6 TSD4-8	YCK4-1 TSD4-1 YCK4-2 YCK4-4 TSD4-4 YCK4-5 YIG4-6 YCK4-6 YCK4-7 TSD4-7 YIG4-8 YCK4-8	YIG3-9 YCK3-9 YSC3-9 TSD3-9 TKC3-9 YIG3-10 YCK3-10 YSC3-10 THP3-10 YIG3-11 YSC3-12 YCK4-3 YIG4-7 YSC4-7 TKC4-8	TKC4-2 TKC4-3			
8					YSC4-1 YSC4-3		TKC4-6 TKC4-7	

Fig. 10.8. (continued)

Comparison of ART and Kohonen Networks

As mentioned previously, ART was feasible for classification of community data (Carpenter and Grossberg 1987). The Kohonen network and ART have their own advantages in groupings. Based on our experiences with community data, ART appeared to be more feasible in extracting information for discovering patterns than the Kohonen network on certain conditions. As shown previously, the Kohonen network was able to decipher patterns in the community data (Fig. 10.3). In this case, however, the data were based on densities of species that had high noise levels, i.e. many species with low or zero density. With the data for ART the species data were summed to the selected taxa, and the data were arranged to be smooth. In this type of data without much noise, ART tended to perform better for grouping community samples (Chon et al. 2000c).

Information extraction by ART could be in fact confirmed by subsequent training on ART's weights by the Kohonen network. The Kohonen network efficiently extracted information of the weights produced from ART and produced a 2 dimensional map (Fig. 10.8b). The mapping by the Kohonen network correspondingly reflected the characteristics observed at the classification results from ART (Fig. 10.8a). The polluted sites - THP and TCL - were closely located and separated from the less-polluted sites. Communities collected from the medium pollution in TKC formed small groups widely dispersed on the map, while the less-polluted sample sites were generally divided according to

topographical conditions. This indicated that the features of the input data from ART were accordingly conveyed to the Kohonen training. The reversed process, producing the weights from Kohonen network first then training the weight subsequently by ART, was not in general effectively conducted in comparison with the ART-Kohonen process. However, it is still early to generalize that ART is more effective in community data than the Kohonen network. Various other factors are involved in formation of community groupings, and further investigation is required.

The Kohonen network has another advantage of visual presentation. It projects the data feature on a map in a reduced spatial dimension (conveniently 2 or 3) as shown in Fig. 10.3 and Fig. 10.8b. ART neurons were not structured spatially (Fig. 10.8a). The output results on the Kohonen network, then, are more comprehensible in characterizing the conformation of neurons.

Large Scale Classification

For the sustainable ecosystem management, surveys in large-scale spatial and time domains are frequently required. For establishing strategies for land management or water quality control on the national basis for example, comprehensive understanding of the total community pattern is necessary. For fulfilling the goal of the long-term study on a large area, a steady and consistent sampling under well-defined survey planning is necessary, and this project consequently produces a large amount of data. The Kohonen network has the advantage of organizing a large-scale data.

Fig. 10.9 demonstrates the possibility of a large-scale grouping. A two dimensional map was produced after the training with the Kohonen network on the sampled communities of benthic macroinvertebrates collected in streams of South Korea for twelve years from 1984, which had been published in the 14 papers from 23 tributaries in the major river system (Chon et al. 2000a). The communities appeared to be grouped according to the river systems (e.g., Han River, Somjin River, etc). The Han River have been most extensively surveyed, and the communities collected from the Han River were further sub-grouped according to the degree of pollution on the map (Chon et al. 2000a). If newly sampled communities are given to the network, they would be conveniently recognized as mentioned previously, These processes of visual presentation of large-scale data and recognition of new data sets could be efficiently used for diagnosing ecological status of the surveyed area for a long time for sustainable ecosystem management.

	0	1	2	3	4	5	6	7	8
0		41, 58	35, 37, 38, 39, 52, 67		77, 79		(Masan River) 91, 92, 94, 99, 100, 108	102	
1	97		51, 54, 59		46, 48	45, 57, 66	96, 103		109
2			49, 50	28	(Somjin River) 8, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 29, 31, 47, 111, 113	23			53, 55, 60, 80
3	32		24, 25, 26, 30, 33, 34	88	56, 124	(Han River) 0, 1, 69, 70, 71, 72, 73, 74, 75, 83, 107, 114, 115, 117, 118, 119, 120, 121, 122, 123	89, 90, 93, 106		
4				87	(Han River) 2, 3, 4, 5, 6, 7, 9, 110, 112, 116				
5	44		36, 81	40, 68, 84, 85, 86					95, 101, 104
6	61	27, 64		105	63	62, 76			
7						65, 82	42, 78		
8	43		98						

Fig. 10.9. Mapping of benthic communities collected at the large scale in South Korea for 12 years form 1984 by the Kohonen network after training. (From Chon et al. 2000a).

10.2.2 Grouping Community Changes

Since community develops on time domain, either in stressful or in favorable conditions, groupings of community "changes" is necessary for the comprehensive perspective on stream ecosystems. Especially in aquatic ecosystems, where communities are easily affected by disturbances caused by various natural and anthropogenic agents (Sladecek 1979; Hellawell 1986), it is important to pattern community changes in response to disturbances. However, it is not an easy task to classify community changes, and fewer studies have been conducted on this topic. Legendre et al. (1985) and Legendre (1987) discussed classifying communities in temporal domain, utilizing ordination and segmentation techniques in multivariate data series. Similar to the case of static classification, the conventional statistical methods however, are limited in analyzing complex data for community changes.

Artificial neural networks, however, could be further implemented to grouping "changes" in community. A combined model of artificial neural networks was utilized in this case. The sampled data for community changes were trained with the two processes through Adaptive Resonance Theory (ART; Carpenter and Grossberg 1987) and the Kohonen network (Kohonen 1989). The schematic diagram of the combined network is presented in Figs. 10.7a and 10.7b. Initially all the community data for one-time sampling from field were trained by ART

(Fig. 10.7b) as mentioned previously and subsequently the weight produced by ART were mapped by the Kohonen network (Fig. 10.7a) (Chon et al. 2000c).

As explained before, weights produced by ART preserved conformational characteristic of input data for each sampling time through training (Fig. 10.8a). Since one-month sampling data were accordingly characterized by weights in the Kohonen network (Fig. 10.8b), it was supposed that, if the weights for previous months were appended to the target month, they would also efficiently represent changes in community during the specified period. The weights trained for one month in ART were combined sequentially for a certain period, and were given to the Kohonen network as inputs. If community changes in three months are to be patternized and March is the target month for training, for example, weights for the two previous months were appended in front of March (i.e. January - February - March). In total T x n weights were used for inputs where T and n respectively represented for the number of months and that of variables for training as inputs for ART (Figs. 10.7a and 10.7b). This was similar to creating a window of a specified input period (three months in this case) and scanning all through the target sampling times through the survey period. The detailed process could be referred to Chon et al. (2000c).

The self-organizing Kohonen network with the array of T x n artificial neurons maps the data feature in a reduced dimension as previously demonstrated. In this case two-dimensional array with 9 x 9 neurons was used. The weights in the Kohonen network were represented as $w_{k,(mj^*i)}(t)$. Since j^* was determined as a winner in ART, and all the winner nodes were selected for input to the Kohonen network, j^* was set to a constant for training. Among designating digits for input node, sampling month in the sequential period, m, and input node for ART, i, were varied in this case. When the input vector was sent through the network, each output neuron, k, computed the summed distance, $d'_k(t)$, between input vector and weights, and subsequently training is conducted in the similar manner explained previously for the Kohonen network.

The field data used for ART training (Fig. 10.8a) was also provided as input to the network for grouping community changes. Figs. 10.10a and 10.10b show the mappings for two and three months after the training by the Kohonen work. The trained results showed general characteristics as observed in the results from one-month samples (Fig. 10.8b). Grouping was mainly based on pollution levels and topographical conditions. In the two-month sequences (Fig. 10.10a), several large groups appeared. A large number of samples from polluted sites of TCL and THP were grouped together (Group A) at neuron (6, 0). This was also the case in the one-month mapping (neuron (0,3) in Fig. 10.8b). Many sample communities from YCK and TSD formed another large group (Group B) at neuron (3, 7). Communities collected at YIG in the early part of 1994 also formed a group (Group C) at neuron (6, 3). Communities from TKC were spread on the map with small groups, similar to Fig. 10.8b. In contrast to one-month sampling, however, a slight difference was observed in the two-month map (Fig. 10.10a). Sample data from TSD were absorbed into Group B in the two-month map. Communities collected from YSC, which were mostly located close to, or inside the group

a)

	0	1	2	3	4	5	6	7
0			RHP3-10 THP3-11 TCL3-12		YSC4-2 YSC4-4	TKC4-4	TCL3-10 THP4-1 THP4-2 THP4-3 THP4-4 THP4-5 THP4-6 TCL4-2 TCL4-3 TCL4-4 TCL4-5 TCL4-6	
1							TCL3-11 TCL4-6 THP4-6	
2	TSD3-11 TSD3-12				TKC3-11 YCK3-12		TKC4-1	
3							YIG4-1 YIG4-2 YIG4-3 YIG4-4 YIG4-5 TKC4-5	
4	TKC4-7 TKC4-8	TKC4-6	TKC4-2		TKC4-3			
5							YSC4-5	
6								
7	YSC4-1		YSC4-6 YSC4-6	YIG3-10 YIG3-11 YIG4-6 YIG4-7 YCK3-10 YCK4-1 YCK4-4 YCK4-5 YCK4-6 YCK4-7 YCK4-8 YSC3-10 YSC4-3 TSD3-10 TSD4-1 TSD4-2 TSD4-3 TSD4-4 TSD4-6 TSD4-7 TSD4-8		TKC3-10 YCK3-11 YCK4-2 YSC3-11 YSC4-8		
8	YIG3-12		YSC3-12	YCK4-3 TSD4-5 YIG4-7				

b)

	0	1	2	3	4	5	6	7
0	THP3-11 TKC4-8			YSC4-3 YSC4-5	YCK3-11 YSC3-11	YIG3-12 YCK3-12 YSC4-2 YSC4-4	TKC3-11	
1	YIG3-11 YIG4-6 YIG4-7 YIG4-8 YCK4-1 YCK4-2 YCK4-3 YCK4-4 YCK4-5 YCK4-6 YCK4-7 YCK4-8 YSC4-1 YSC4-8 TSD3-11 TSD3-12 TSD4-1 TSD4-2 TSD4-3 TSD4-4 TSD4-5 TSD4-6 TSD4-7 TSD4-8	YSC4-7					YSC3-12	YIG4-1
2	TKC4-6 TKC4-7				THP4-2 THP4-3 THP4-4 THP4-5 TCL4-3 TCL4-4 TCL4-5			
3	TKC4-1		THP4-1 THP4-8 TCL4-2 TCL4-8				THP3-12 RCL4-1	
4								
5				TKC4-3				
6								TCL3-11 TCL4-6 THP4-6
7	THP4-7 TCL4-7			TCL3-12				
8								

Fig. 10.10. Mapping of benthic macroinvertebrate communities collected in the Suyong River when the temporal variations were trained by the Kohonen network. The alpha-codes for the name of sample communities are explained in the caption of Fig. 10.8. (From Chon et al. 2000c) a) two month training. b) three-month training.

	0	1	2	3	4	5	6	7	8
0	YCK4-11								
1	YIG4-9 YIG4-10 YIG4-11 YCK4-9 YCK4-10 YSC4-10						YSC4-9	YSC4-11	
2									
3									
4									
5									
6									
7									
8									

Fig. 10.11. Recognition of newly collected benthic macroinvertebrate communities to the trained Kohonen network in the period of three months. The alpha-codes and numerical digits designating the name of sample units are explained in the caption of Fig. 10.8. (From Chon et al. 2000c).

mainly consisting of YCK (neurons (2,7) and (3,7) in Fig. 10.8b), tended to drop off at group B in the two-month map in some cases. However, the YSC communities did not move far way from Group B (Fig. 10.10a).

In the maps describing community pattern of the period longer than two months, the characteristics shown in the two-month map were generally preserved. Most sample communities in Groups B and C were consistently found inside the groups as the input period increased. Sample communities from TKC and YSC also showed a similar tendency as seen in the two-month map. In Group A, however, the size of samples was gradually decreased. In the 3-month map (Fig. 10.10b), for example, THP4-1, THP4-6, TCL4-2, TCL4-6 were separated from Group A. Detailed discussion and patterns of community development longer than three months could be referred to Chon et al. (2000c).

Once training of community changes was completed, recognition for a new input data by the network was possible. Benthic communities collected at the sites from Suyong Stream (YIG, YCK, and YSC) from September to November in 1994 were used for recognition. Initially the new input data were given to ART and weights were updated as explained before. The updated weights were then arranged sequentially for a given period (for three months in this case), and were subsequently provided to the trained Kohonen network for recognition (Fig. 10.11). Generally most of input data were recognized to belong to Group B (See neuron (0,1) in Fig. 10.10b for the three-month map), which was the main group of community formed from the Suyong Stream. The recognized results were generally expected patterns from the field experience.

As mentioned previously ART was better in classifying the smooth community data, while the Kohonen network was more feasible in grouping data with many zeros. The output results, however, were more visually comprehensible with the Kohonen network in characterizing the conformation of neurons in spatial

dimension (conveniently 2 or 3) (Fig. 10.10). This was the reason that we first used ART, and then implemented the Kohonen network for training community changes in this study. With the combined use of the two unsupervised neural networks it was possible to patternize temporal variations in community data.

10.3 Prediction of Community Changes

10.3.1 Multilayer Perceptron with Time Delay

In the previous section, grouping of communities was presented. Through grouping techniques, however, actual densities of taxa in community could not be provided. Prediction of actual values in temporal development of communities, however, is a major concern in ecosystem management. Especially in aquatic ecosystems, where communities are easily affected by disturbances caused by various natural and anthropogenic agents, it is important to predict how communities would develop in response to changes in water quality. They would develop either progressively with further disturbances, or regressively in recovery from pollution (Sladecek 1979; Hellawell 1986). It is essential to predict the future level of community abundance for monitoring as well as for assessing ecological status of the target ecosystem.

As previously mentioned, data for community dynamics, however, are complex and difficult to analyze. In temporal patterning in ecology, artificial neural networks has been effectively implemented in estimating time development of populations and communities such as flowering and maturity of soybean (Elizondo et al. 1994), algal bloom (e.g., Recknagel et al. 1997; Recknagel and Wilson 2000), dynamics of animal population (Stankovski et al. 1998), and grassland community development (Tan and Smeins 1994). However these models were in most cases applied in static terms; the time of input and output were the same. Recently, attention also has been given to dynamic neural networks (e.g., Kung 1993; Giles et al. 1994; Haykin 1996). Wray and Green (1994) reported that artificial neural networks could be utilized for investigating parameters in non-linear dynamics, and dynamics of ecological data were patterned and predicted by Boudjema and Chau (1996) on sets of univariate time-series data of tree-ring thickness

To pattern relationships between different time events of community changes, initially a well-known multilayer perceptron was utilized as a nonlinear predictor with the backpropagation algorithm (Wray and Green 1994; Haykin 1994) (Fig. 10.12a). The architecture is multiplayer perception, however input and output data were provided with time delay. The input vector is defined in terms of the past samples, $X(t\text{-}1)$, $X(t\text{-}2)$, . . ., $X(t\text{-}q)$, where q, prediction order, is the number of the total delays. The current data, $X(t)$, was given as matching output.

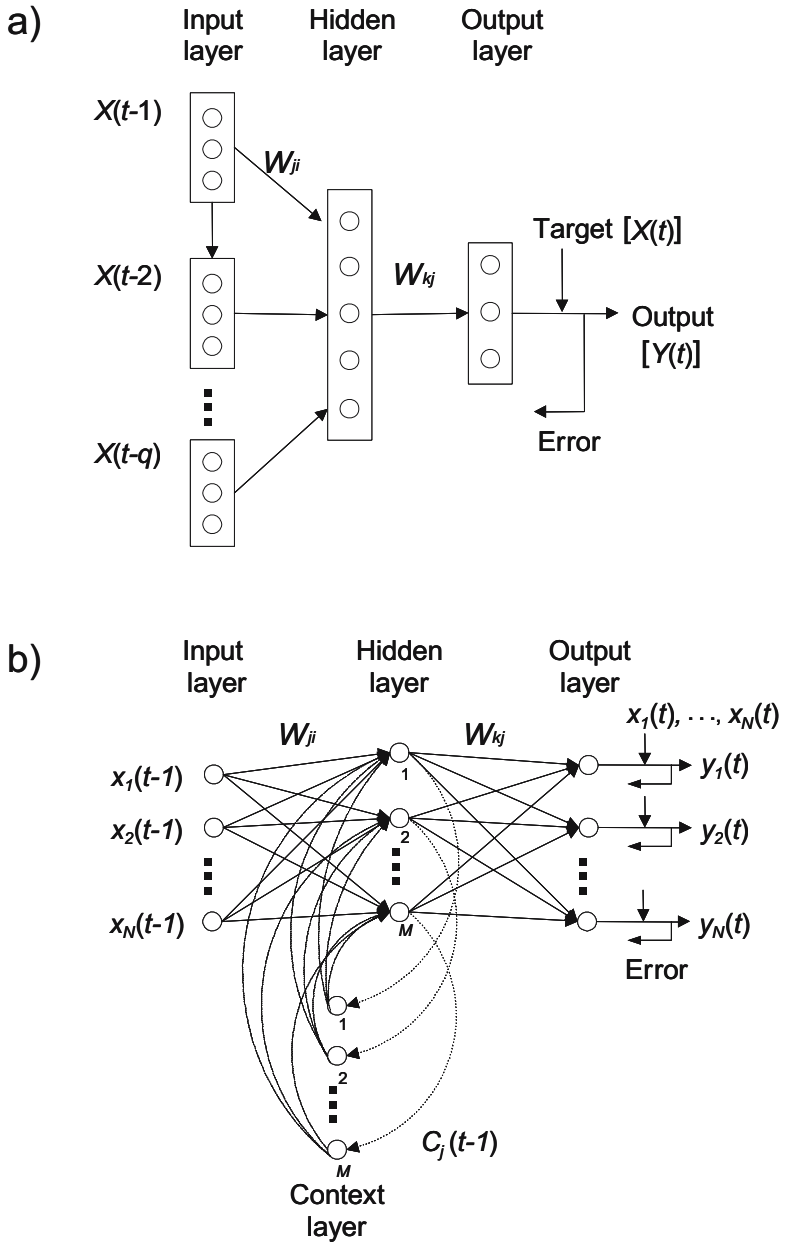


Fig. 10.12. The architecture of artificial neural networks with temporal patterning. (From Chon et al. 2000b). a) time delayed multilayer perceptron. b) Elman type recurrent neural network (RNN).

For field data, samples were collected in a relatively short distance within 200 meters in the Yangjae Stream, a tributary of the Han River. The Yangjae Stream is located in the Seoul metropolitan area, on the middle part of the Korean Peninsula, and is highly polluted with poly-saprobity. In selecting data, attention was given to taxa more frequently and abundantly collected while the data for rare species were not included for training. *Chironomus* sp., *Orthocladius* sp., *Cricotopus* sp., *Limnodrilus* sp. and *Erpobdella* sp. were chosen. The first three species are Chironomidae, while the fourth and fifth species belong to Oligochaeta and Hirudinea, respectively. These selected Genera occurred consistently at the study sites during the survey period. Data collected from March 1996 to March 1998 were used for the learning process, and a portion of samples were set aside for recognition (Chon et al. 2000b).

Densities of the selected 5 Genera in sampled communities were provided as data sets for inputs with 1 – 5 time the delays, i.e., q = 1, 2, …, 5. With each delay, input nodes were correspondingly added. For example, if 5 Genera were introduced with 2 time delays, 5 × 2 = 10 nodes were assigned for each input. The input layer was subsequently interconnected to the hidden layer. Eight to thirty nodes were used in the hidden layer. The number of nodes in the hidden layer was determined based on experiences on obtaining convergence in training. The number of output nodes was 5, equal to the number of selected Genera. Similar to the static implementation, the internal state of the network, $NET_{p,j}$, was obtained by linear summation of products of weights and output values of nodes in the hidden layer over time. Subsequently, these values were adjusted in a nonlinear fashion, logistic function in this case, to produce outputs, $Y(t)_{p,j}$, as follows (Wasserman 1989; Zurada 1992, Haykin 1994):

$$NET_{p,j} = \sum_{i=1} x_{p,i} w_{p,ji} \tag{10.6}$$

$$Y_{p,j} = \frac{1}{1 + \exp(-\lambda NET_{p,j})} \tag{10.7}$$

where $Y_{p,j}$ is activation of neuron j for pattern p, $x_{p,i}$ is output value of the neuron i of the previous layer for pattern p, $w_{p,ji}$ is weight of the connection between the neuron i of the previous layer and the neuron j of the current layer for pattern p, and λ is activation function coefficient (e.g., 1.0 in this study).

The output $Y(t)$ of the multilayer perceptron was produced in response to the input vector, and was equivalent to the one-step prediction for the future development. Subsequently actual data at time t, $X(t)$, were provided as the target and the difference between $Y(t)$ and $X(t)$ was measured and propagated backward for adjusting weights in the usual manner of the backpropagation algorithm (Rumelhart et al. 1986). Weights at output neurons were updated as follows :

$$\delta_{p,j} = Y_{p,j}(1 - Y_{p,j})(d_{p,j} - Y_{p,j}) \tag{10.8}$$

$$\Delta w_{p,ji}(t+1) = \eta \delta_{p,j} Y_{p,j} + \alpha \Delta w_{p,ji}(t) \tag{10.9}$$

$$w_{p,ji}(t+1) = w_{p,ji}(t) + \Delta w_{p,ji}(t+1) \tag{10.10}$$

where $d_{p,j}$ is desired output of node j for pattern p, η is training rate coefficient, and α is momentum coefficient. Weight updating at the hidden layers is similar to processes at the neurons of the output layer.

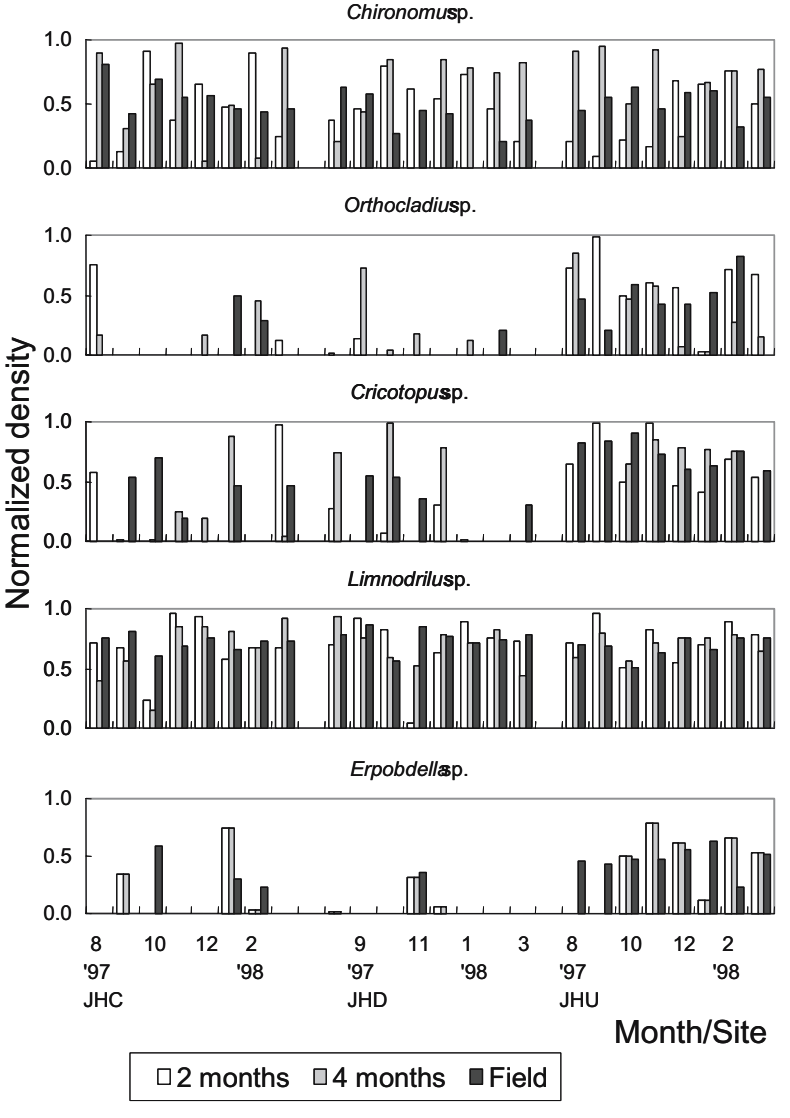


Fig. 10.13. Examples of field data and predictions in densities of selected Genera after recognition by the trained multilayer perceptron with time delay (two and four months) when new data for community development were given as inputs in the Yangjae Stream. (From Chon et al. 2000b).

Detailed process for the backpropagation algorithm could be referred to Rumelhart et al. (1986) and Zurada (1992).

When communities were given as input to the simple multilayer perceptron with the time delay between one and five months, the convergence was generally reached in the iteration of 20,000 - 30,000 under the mean error term of 0.05, which is the sum of square terms of difference in output and target values divided by the number of input patterns. The training rate learning and momentum coefficients were 0.5 and 0.9, respectively. Trained data sets were accordingly matched to the original input data. It appeared that convergence was dependent upon the length of time delay (Chon et al. 2000b). When new data were given to the trained network for recognition, the network was able to make one-step predictions for the following community in time (Fig. 10.13). In general it appeared that the predicted and actual field data were in accord, although some discrepancies were locally observed. The trained results correspondingly reflected the expected development of communities under the influence of various degrees of organic pollution in urbanized streams in a certain time span. However, there were occasions that the degree of correspondences between the actual and predicted data was not high. It was difficult to obtain precise matching in densities between the two data sets. This is understandable that predicting density in "each" taxa of communities in field conditions are generally not easy. Correlation coefficients between the predicted and field data were 0.556 (P<0.0001) and 0.489 (P<0.0001) respectively for the two and four month delays in the Yangjae Stream.

10.3.2 Elman Network

The temporal development of input data could be revealed by the recurrent neural network. The Elman type network (Elman, 1990), one of the most well-known dynamic models in partially connected recurrent learning, was implemented for learning community dynamics (Chon et al., 2000b). The architecture of Elman type recurrent neural network (RNN) is basically similar to the multilayer perceptron except the composition of the hidden layer (Fig. 10.12b). However, hidden layer embodies another context layer for implementing recurrence. Recurrence implies that the state of network depends on current input and its own internal state on the previous cycle. In this case, the hidden layer has recurrence and its own internal state is represented through the context of the hidden layer. The number of nodes at the input and output layers was 5, and 30 neurons were used for the hidden and context layers.

In the input layer, community data for selected Genera, $x_I(t\text{-}1)$, were given as external input. Concurrently, output values from the hidden layer for the previous cycle are also provided as internal inputs to the hidden layer as $C_I(t\text{-}1)$. Initially, some small random numbers are used for the internal inputs. The group of $x_I(t\text{-}1)$ and $C_I(t\text{-}1)$ consist of the total input for the hidden layer, $z_I(t)$. The sum of linear

combination of weights and inputs, $I_j(t\text{-}1)$, is subsequently adjusted in a nonlinear function such as $C_l(t\text{-}1) = f(I_j(t\text{-}1))$. The input process could be summarized as follows (Hecht-Nielsen, 1990):

$$z_l(t) = \begin{cases} x_l(t) & \text{if } 1 \le l \le N \\ C_l(t-1) & \text{if } (N+1) \le l \le L \end{cases} \tag{10.11}$$

where l = 1, 2, …, L, $L = N$ (number of input nodes) + M (number of hidden nodes), $x(t)$ is external input, and $C(t)$ is context input.

$$I_j(t) = \sum_{l=1}^{L} w_{jl} z_l(t) \tag{10.12}$$

$$f(I_j(t)) = \frac{1}{1+\exp(-\lambda I_j(t))} \tag{10.13}$$

$$C_l(t) = f(I_j(t)) \tag{10.14}$$

The net output in the output layer is determined by the summation of the linear combination of weights and values produced from the hidden layer. As a usual process in artificial neural networks, this is subsequently adjusted with a nonlinear function, logistic equation in this case, to produce output values for t as $y_k(t)$. These output values are in turn compared with actual field data, $x_l(t)$. Weight adjustment is conducted in the same way as it is determined in the backpropagation algorithm. The difference between desired output and internal output was calculated, and subsequently was backpropagated through the hidden layer down to the context and input layers.

When communities were trained with the Elman network (Fig. 10.14), convergence was also achieved and its learning efficiency generally appeared to be higher than that by the simple multilayer perceptron (Fig. 10.13). The mean error term was apparently lower than that shown in the training by the previous multilayer perceptron (Chon et al. 2000b).

When new data were given to the trained network for recognition, the network was able to predict community abundance for the next month (Fig. 10.14). It appeared that the predicted and actual field data were generally in accord, better than in the case of the training with the multilayer perceptron. Correlation coefficients in the data from recurrent neural networks were higher than those from the multilayer perceptron with time delay, by showing 0.675 (P<0.0001) in this case. This demonstrated that the training by recurrent network is more efficient than in the training by the simple multilayer perceptron in their implementation to changes in this type of community data.

Another advantage with the forecasting for each taxon is that it could assist to characterize community changes. Even if the predicted data were not in accord with field data, for example, it would still give information to investigate the status of communities. Since neural networks represent average effects with this

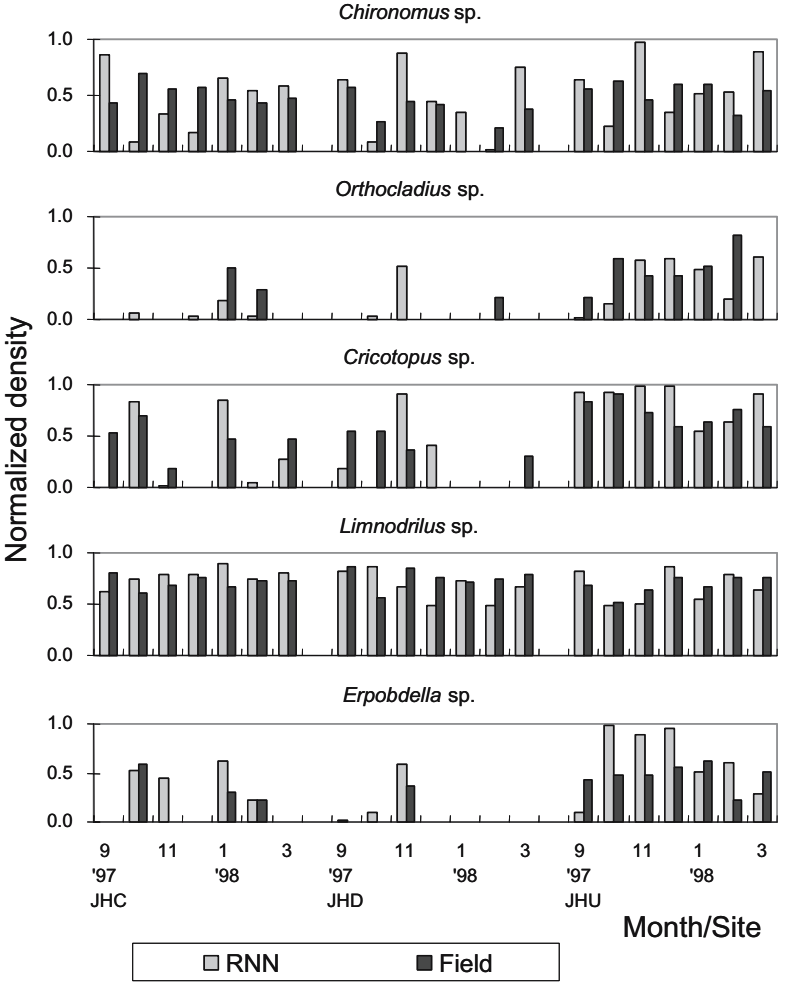


Fig. 10.14. Examples of field data and predictions in densities of selected Genera in communities of benthic macroinvertebrates after recognition by the Elman type recurrent neural network (RNN) when new data for community change were given as inputs in the Yangjae Stream. (From Chon et al. 2000b).

type of training, more frequently appearing taxa would have more chance to be patterned in the training. Then the mismatching between the actual and predicted data may suggest occurrence of some disturbances in communities of new data. The forecast of 'changes-in-density' could effectively pinpoint the shifting of ecological status in communities (Chon et al. 2000b).

10.3.3
Fully Connected Recurrent Network

From the previous training, recurrent property in artificial neural networks was shown to be effective in extracting information on the time development of community. A fully connected recurrent network was further implemented for patterning community dynamics under the scheme of real-time recurrent learning (Willians and Zipser 1989). This real-time recurrent network (RTRN) has been characterized as containing hidden neurons and allowing arbitrary dynamics, in comparison with other recurrent networks such as the Hopfield network (Hopfield 1982). The RTRN is especially capable of dealing with time-varying input or output through its own temporal operations (Haykin 1994).

The RTRN consists of N neurons with M external input connections (Fig. 10.15a). The external input vector of community data $\boldsymbol{x}(t)$ of size M is applied to the network at a discrete time t. Let $\boldsymbol{y}(t)$ denote the corresponding vector of size N of individual neuron outputs produced one step later at time t. The N neuron outputs at the upper processing layer consist of M neuron outputs and $(N\text{-}M)$ hidden neuron outputs. The input vector $\boldsymbol{x}(t)$ and the one-step delayed output vector $\boldsymbol{y}(t\text{-}1)$ are concatenated to form the vector $\mathbf{u}(t)$ of size $(M+N)$ whose ith element is denoted by $u_i(t)$ (Haykin 1994). In total, an N by $(M+N)$ recurrent weight matrix is formed.

The net internal activity of neuron j at time t is as follows:

$$v_j(t) = \sum w_{ji}(t) u_i(t) , \tag{10.15}$$

where $v_j(t)$ is $x_j(t)$ if j denotes the external input, and $y_j(t\text{-}1)$ if j denotes the neuron for outputs. $w_{ji}(t)$ is the weight between the input and the hidden layers. At the next time step $t+1$, the output of neuron j is computed by passing $v_j(t)$ through the nonlinearity $\psi(\cdot)$ (logistic function in this case), resulting in the following (Haykin 1994):

$$y_j(t) = \psi(v_j(t)) . \tag{10.16}$$

The backpropagation algorithm (Rumelhart et al. 1986) was further implemented in this study. The real time recurrent learning handles weight feedback in the real time process and allows faster convergence in recurrent learning. The detailed algorithm could be referred to Williams and Zipser (1989).

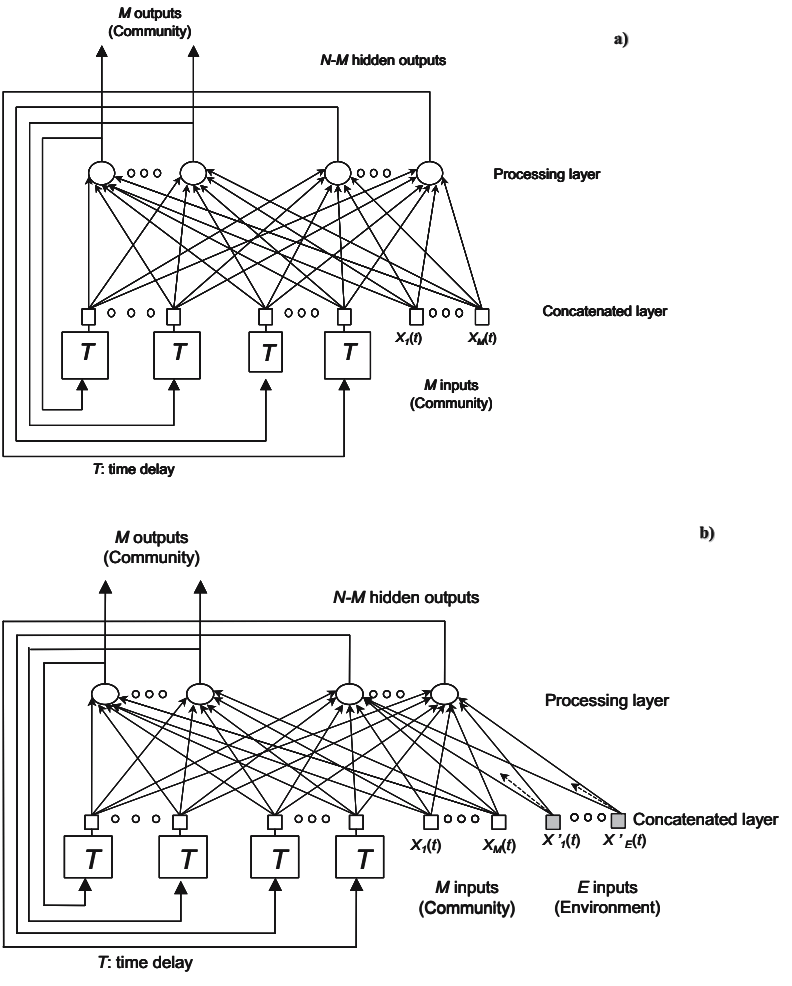


Fig. 10.15. A diagram of the real-time recurrent network for patterning community changes in benthic macroinvertebrates in streams. (From Chon et al. 2001). a) input data: community. b) input data: community plus environment.

The learning rate and the momentum coefficient were 0.3 and 0.7, respectively. In this study, 7 neurons were used for community data for external inputs, and 13 neurons for hidden nodes. The error term was the sum of the difference between the output and the target data for all nodes (selected taxa) for all patterns (sample sites), and the criterion of the error term for allowing convergence was 0.006. The data for the previous three months were given as the input in a sequence with recurrent feedback, while the data for the fourth month were provided as the matching output.

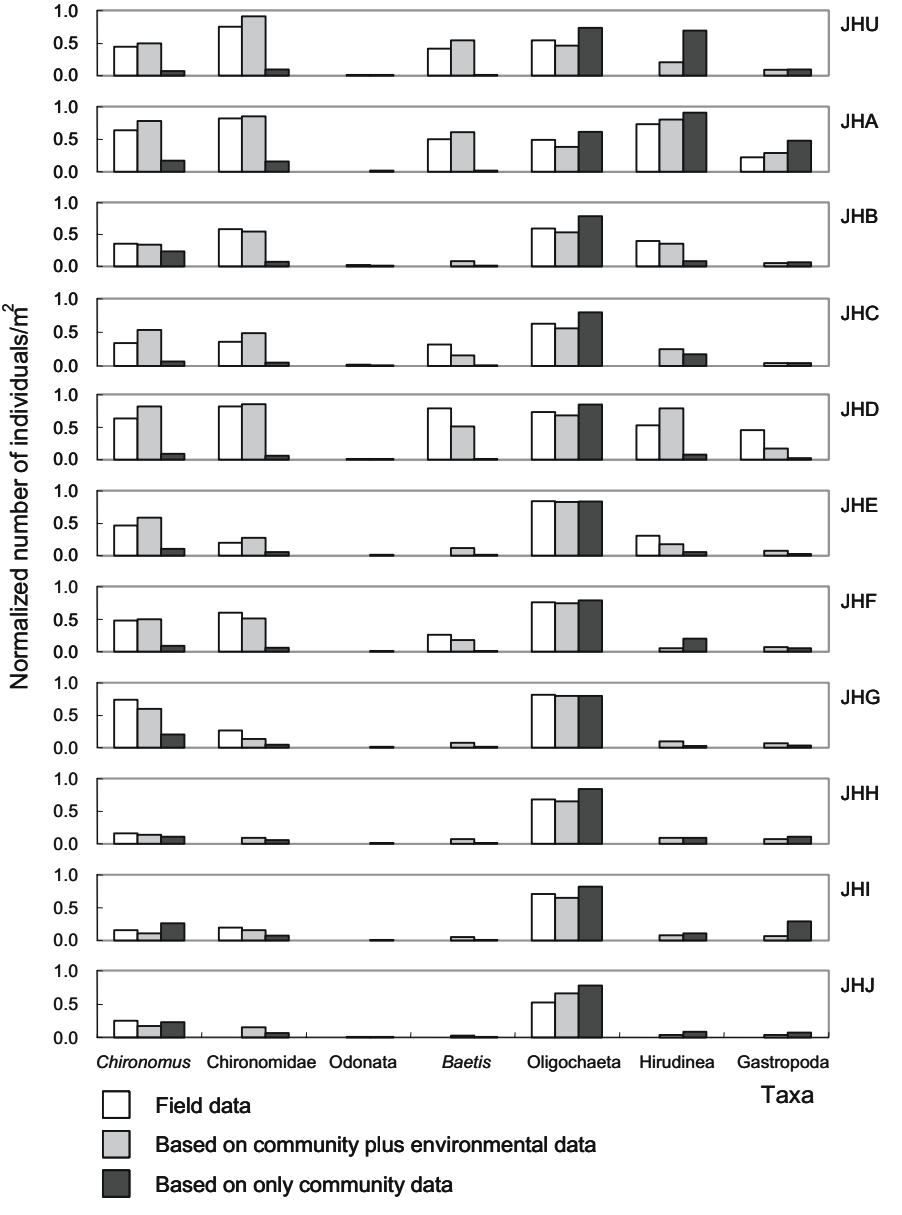


Fig. 10.16. Field and predicted data after training with the real-time recurrent network based on community plus environmental data and only community data. (From Chon et al. 2001). a) July 1997. b) November 1997.

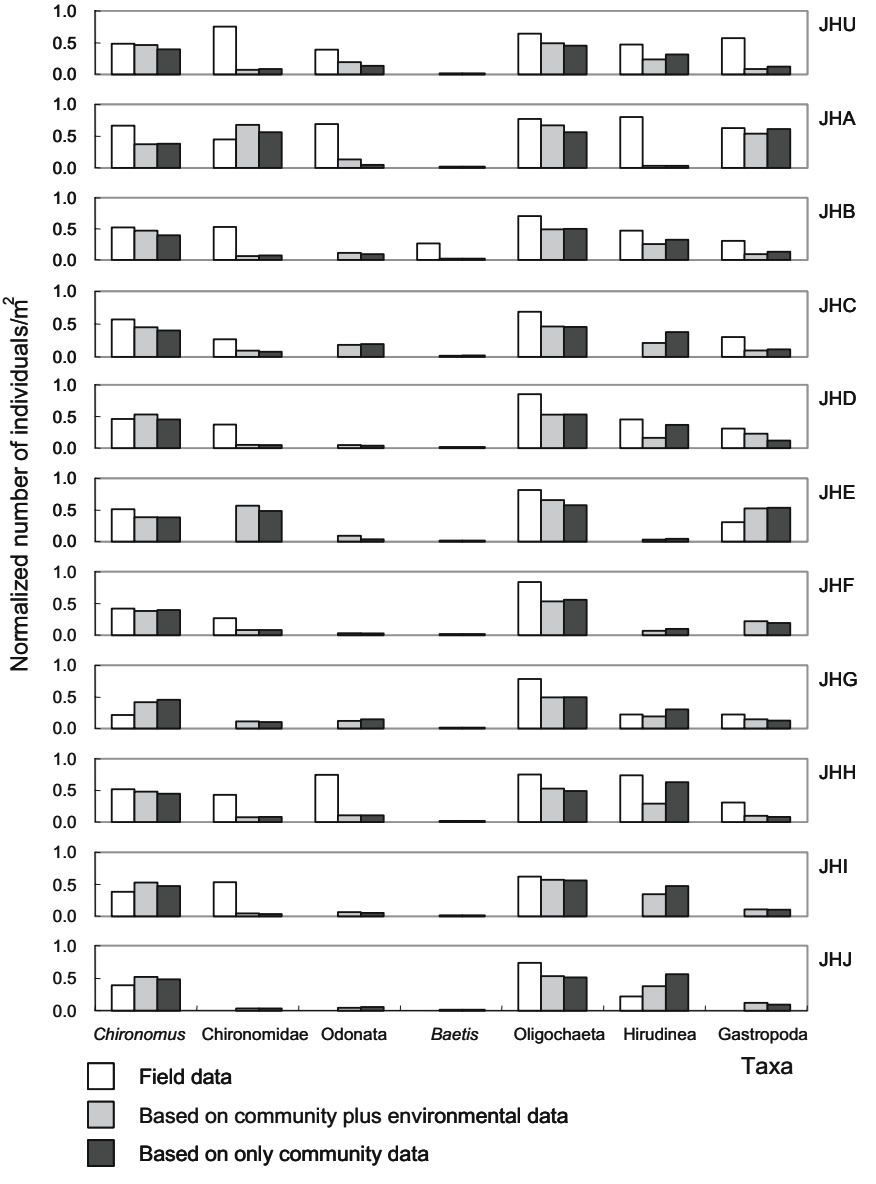


Fig. 10.16. (continued) b) November 1997.

The field data were benthic macroinvertebrates collected at sample sites located on a reach within 200 m of the Yangjae Stream, a tributary of the Han River in Korea (Fig. 10.1). Data collected from April 1996 to March 1997 were used for the training set, while data collected from April 1997 to March 1998 were used as

new data for testing the trained network.

When community data were trained with the RTRN, convergence was usually reached between the 5,000th and the 10,000th iteration. The training sets were well in accord with the matching output. In order to verify the predictability of the trained network, we further provided new community data from April 1997 to March 1998. Figs. 10.16a and 10.16b show examples of comparing field data with the predicted data in different seasons. Generally, dominant taxa such as Oligochaeta, *Chironomus*, and Chironomidae showed good matches between the predictions and the field observations. Pearson's correlation coefficients (Zar, 1984) between the predicted data and the field data ranged from 0.55 (F=34, P<0.001) to 0.80 (F=9.0, P<0.001). In the similar condition, the data evaluated by the Elman network showed correlations coefficients usually ranging between 0.3 – 0.4 on these types of community data (Chon et al. 2000b).

10.3.4
Impact of Environmental Factors Trained with the Recurrent Network

Revealing the impact of environment on communities is essential for finding causality of community response to disturbances. The impact of environment could be judged by prediction of community abundance corresponding to variations of environmental factors, and many researches have been conducted on measuring the impact of environments in community based on the sensitivity analysis (e.g., Dimopoulos et al. 1995; Scardi 2000; Recknagel and Wilson 2000; Salvador et al., 2001). In this case, however, the time setting has been usually static.

As mentioned previously, however, the time factor is one of the key issues in community dynamics especially in the context of regressive or progressive community changes. In this case, in order to emphasize time-dependency in community data, we trained changes in community and environment in the scheme of recurrent training. The previously mentioned the fully connected recurrent network, RTRN, was modified to accommodate environmental factors, but, unlike the community data, neurons accepting environmental factors did not have recurrence feedbacks (Fig. 10.15b). In concurrence with the input of biological data, the corresponding sets of environmental data were given to the modified RTRN, producing, through the connectivity of the network, continuous, independent effects on determining community abundance. In addition to 7 neurons for community data for external inputs and 13 neurons for the hidden layer, 4 neurons were used for receiving environmental factors separately.

As for environmental data (E), monthly observations of water velocity and depth, amount of sedimented organic matter, and volume of substrates smaller than 0.5 mm were provided to the network. The relationships between the environmental data and community dynamics were successfully extracted, and predictability was greatly increased when the data were trained during a period of strong environmental influences (e.g., flooding). The predicted data by the network trained with the community plus environmental data were distinctively

closer to field data in July 1997 than the predictions with the community data alone as previously mentioned (Fig. 10.16a). In July 1997, the correlation coefficient between the predicted and the matching field data was 0.94 when the network was trained with the input of the community plus environmental data. In contrast, the correlation coefficient was 0.55 when only the community data were given to the network. High densities of chironomids were present in the Yangjae Stream during the flooding season of the training period. This relationship between precipitation and the occurrence of chironomids was reflected in the network trained with the community and the environmental data. As shown in Fig. 16a, the predicted densities of Chironomidae and *Chironomus*, trained with the community and the environmental data, for example, well matched the field data, while the network trained only with the community data consistently underestimated the densities of Chironomidae and *Chironomus* (Chon et al. 2000b).

However, predictions based on environmental data were not always superior to predictions based only with the community data. In seasons without strong environmental effects, the predictability seems to be similar between these two types of training. For example, in the prediction of the community data of November 1997 in winter (Fig. 10.16b), the respective Pearson's correlation coefficients were 0.60 and 0.59 for the training based on the community plus environmental data, and that based only on the community data. During the training period there were no serious environmental disturbances occurred in this case. When the effects of relationships on training factors were complex environmental, influences may be negatively related. The detailed explanation could be referred to Chon et al. (2001).

It is also useful to investigate causality relationships that how environmental factors influence community dynamics through the sensitivity analysis. We conducted sensitivity analyses on the recurrent network so that the varying impact of environmental factors could be revealed on community changes. Variation around the mean value (ranging +50 % and –50 %) was provided to each input value of the environmental variables. For the simplicity of the sensitivity analysis of this recurrent neural network, variation term was given only to the input of the last month.

In terms of different training periods and selected taxa, the sensitivity tests effectively showed important environmental variables in determining community changes. For the data of July 1997, when the flooding occurred during this period used for training, all four environmental variables of organic matter, depth, velocity and substrates (smaller than 5 mm), caused a large variation of communities in a wide range (Fig. 10.17a). For the data of November 1998, which had no strong environmental effects, in contrast, all environmental variables did not produce variations in community dynamics (Fig. 10.17b).

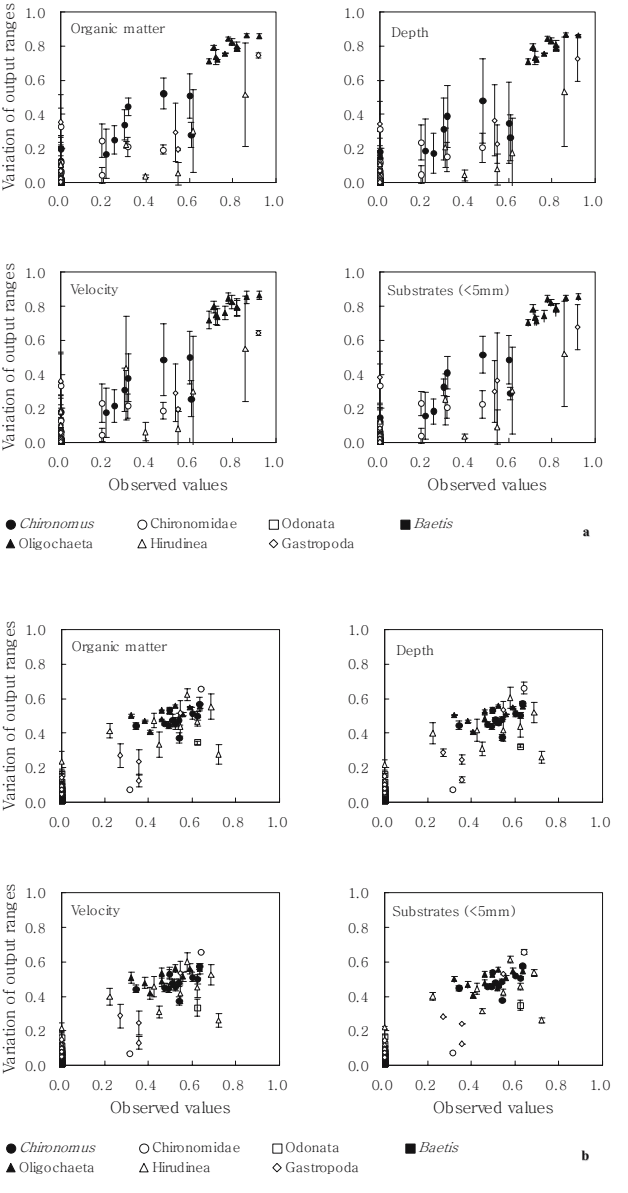


Fig. 10.17. Variation of output ranges in densities of selected taxa in benthic macroinvertebrate community when input values ranging +50 % and –50 % were provided to different environmental variables for the data of July 1997. (Variation was expressed as standard deviation of 11 observations.) (From Chon et al. 2001). a) July 1997. b) November 1997.

The sensitivity tests were also able to show variations for the different taxa of community. Densities of Chironomidae, for example, varied greatly in response to different input ranges. This could be observed for all the variables in the data of July 1997 (Fig. 10.17a). Densities of Hirudinea were also sensitive to environmental variables in the data of July 1997. The higher sensitivity of Hirudinea was also observed in the data of November 1997 without strong environmental effects (Fig. 10.17b). Densities of Oligochaeta, in contrast, were characteristically insensitive to input variables, especially in July 1997. This indicated that, in this field study, the density of Oligochaeta was not greatly affected by environmental variables during the flooding period in comparison with other dominant taxa such as Chironomidae and *Chironomus* sp. (Chon et al. 2000b)

This study examined the feasibility of the recurrent artificial neural network in extracting information out of temporal development. The results showed that the dynamics of sets of multivariate data about communities could be rapidly patterned and forecasted by the network.

10.4
Patterning Organizational Aspects of Community

10.4.1
Relationships among Hierarchical Levels in Communities

Useful ecological informatics resides in community organization, especially in "associations" among different levels in communities such as taxonomical or functional groups. In these associative relationships, the complex community usually develops a hierarchy, which is a good subject for understanding system behavior of the target ecosystem (Allen and Starr 1982; O'Neill et al. 1986). Benthic macroinvertebrate communities in streams usually have clear taxonomic hierarchies and functional groups (e.g., collectors, shredders, etc), and these are essential to verify organizational characteristics in community compositions (Cummins et al. 1973; Cummins 1974). By understanding associative information on community organization, a comprehensive view on ecosystem could be established, and this would help to prepare reliable strategies for achieving the sustainable management of ecosystems. By applying the counterpropagation network, we tried to elaborate the feasibility of artificial neural network to extract information of interrelationships among hierarchical levels in communities of benthic macroinvertebrates in streams (Park et al. 2001a).

The counterpropagation network (Hecht-Nielson 1987) is a type of hybrid model consisting of the two artificial neural networks: the Kohonen self-organizing map (Kohonen 1989) and the Grossberg outstar (Grossberg 1969, 1982) (Figs. 10.18). The network is eventually designed to approximate continuous functional associations between variables, and serves as a statistically

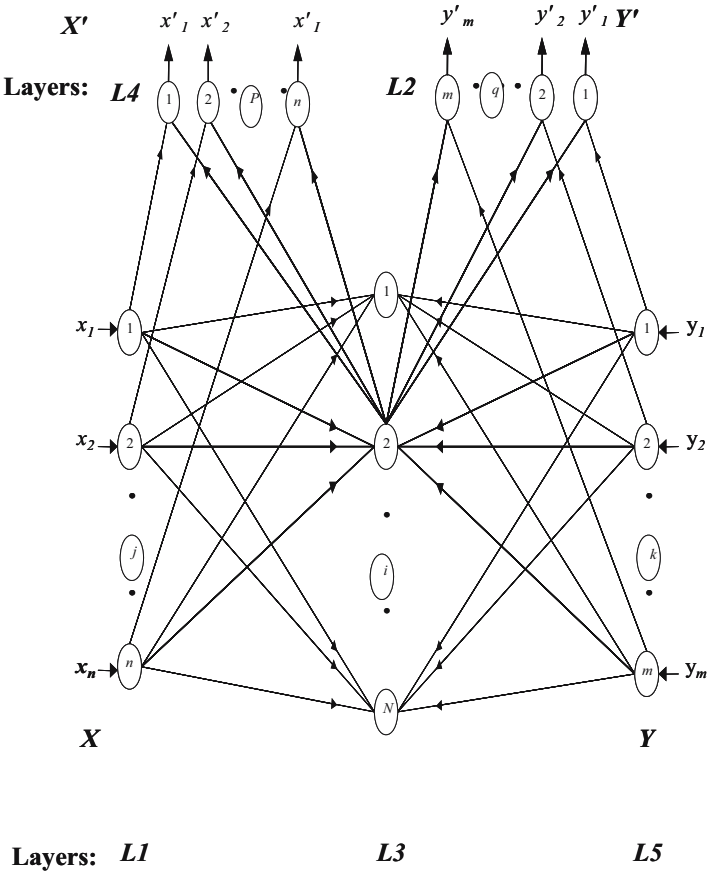


Fig. 10.18. Schematic diagram of the counterpropogation neural network. *n*; number of nodes on input layer, *m*; number of nodes on Grossberg layer, *N*; number of nodes on Kohonen layer. (From Park et al. 2001a).

optimal self-programming lookup-table (Hecht-Nielson 1990). The input data are arranged in two groups according to the hierarchical levels (e.g., *X* for 'Genus/Species' and *Y* for 'Family'). Initially the data for *X* at layer, L1, with *n* nodes are given to the input layer, L3, with *N* nodes, which is the Kohonen layer. At the same time, the data for *Y* at the layer, L5, with *m* nodes are given to the layer L3 (Fig. 10.18). At the layer L3, each node *i* calculates I_i as sum of weights $U_{ij}(t)$ and $V_{ik}(t)$ with two inputs, *X* and *Y*, respectively, as shown in the process 3 in the counterpropagation algorithm (Fig. 10.19). The weights $U_{ij}(t)$ and $V_{ik}(t)$ in Kohonen network were initially given as small random numbers in the process 1 (Fig. 10.19). Among all *N* nodes in the Kohonen layer, the node which has maximum I_{i^*} becomes winner and $Z_{i^*}(t)$ is assigned to be 1 for this winner node

while $Z_i(t)$ for the non-winning nodes remains zero. For the winner and its neighborhood neurons in some distance, the new weights $U_{ij}(t+1)$ and $V_{ik}(t+1)$ were updated by the iterative process as shown in the process 4 in Fig. 10.19. Alpha (α) is a constant determining the learning rate and the values around 0.3 were used in this study. The detailed process in the Kohonen network could be referred to Kohonen (1989) and Chon et al. (1996).

1. Initialize weights for the networks.

2. Present new input.

3. Finding the best matching processing element on Kohonen layer

$$I_i = \sum_{j=1}^{n} U_{ij}(t)x_j + \sum_{k=1}^{m} V_{ik}(t)y_k$$

where x_j and y_k are input data, and U_{ij} and V_{ik} are the weight from input node to Kohonen layer.

4. Update Weights on Kohonen layer.

$$U_{ij}(t+1) = U_{ij}(t) + \alpha(x_j - U_{ij}(t))Z_i$$

where α is a learning rate of forward flow on Kohonen layer, and

$$Z_i = \begin{cases} 1 & \text{if } Ii = Ii^* \ \forall j \quad (i^*; \text{winning node}) \\ 0 & \text{otherwise} \end{cases}$$

5. Present new desired output.

6. Update weights on node p of Grossberg layer.

$$w1_{pk}(t+1) = w1_{pk}(t) + a(y_k - w1_{pk}(t))Z_i$$

where a is a learning rate of forward flow on Grossberg layer. ($w2_{qj}$: similar on node p of Grossberg layer L4).

7. Repeat by going to step 2 until the end of input data.

8. Emit modified desired outputs.

$$y_k' = \sum_{j=1}^{N} w1_{jk}(t)Z_i$$

(x'_p : similar)

Fig. 10.19. Algorithm of the counterpropagation network. (From Park et al. 2001a).

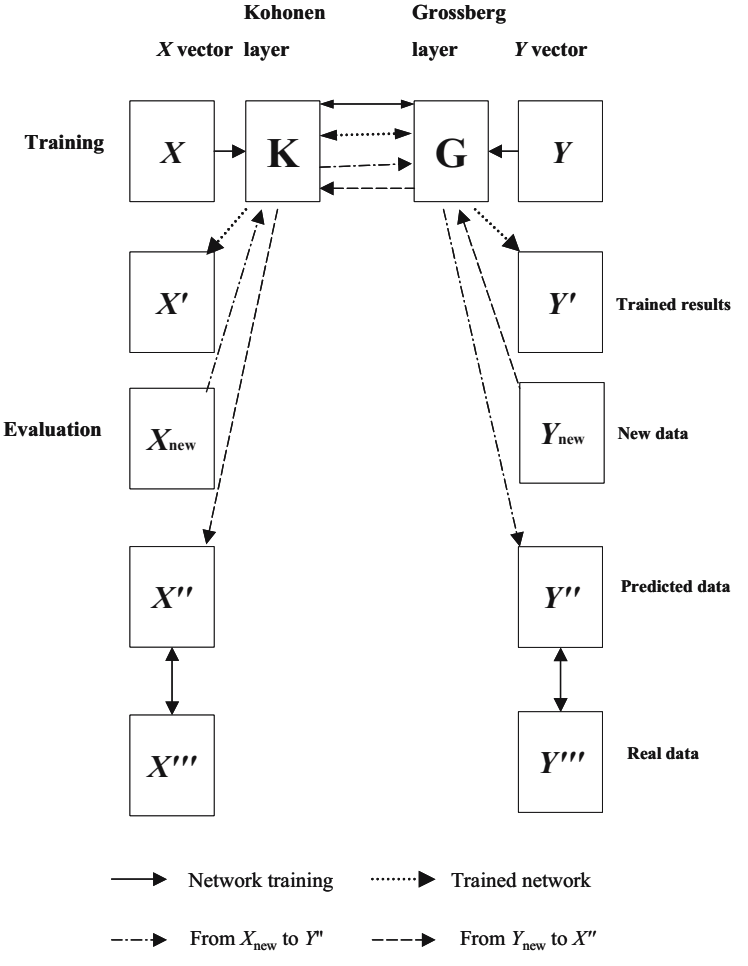


Fig. 10.20. Relational diagrams for training and testing by the counterpropagation network. (From Park et al. 2001a).

After the winner node at the layer L3 was determined, the process was proceeded to the layer L2 and L4 respectively for Y and X, which are Grossberg layers (Grossberg 1969, 1982). At the layer L2, $W1_{qi}$ for Y ($W2_{pi}$ for X), which is the connecting weight between p node of the Grossberg layer and i node (winner) of the Kohonen layer, was updated by the iterative method as shown in the process 6 in Fig. 10.19. Beta (β) (ca., 0.3) is a parameter determining the learning rate. Similar to the Kohonen layer, the weights were initially given as small random numbers. Subsequently the node at the layer L2 produced output, X', by summing the weights connected to all nodes in the layer 3 (Process 8 in Fig. 10.19). Concurrently the nodes at the layer L4 calculated output Y' (Fig. 10.20). By

repeating this process until the weight difference became sufficiently small, the effective information characterizing relations of the two variable sets were preserved in the weights of the network. *Y'* served as trained results with the input of *X*, while *X'* matching to the input of *Y* (Fig. 10.20).

Since the patterns of relation were established between *X* and *Y* in the network through counterpropagation, the variables in the hierarchical levels could mutually respond to each other to new data sets. For example, if newly collected input data set, X_{new}, were given to the trained network, they would produce new recognized data set *Y"* corresponding to X_{new}. This could be compared with the actual field data, *X* "" (Fig. 10.20).

The benthic macroinvertebrate communities in streams monthly collected in the Suyong Stream in the Suyong River in Korea from November 1992 to December 1994 for two years were used for training by the counterpropagation network. As previously mentioned, a wide range of organic pollution was shown in the study area. In order to check community compositions in a limited range in environmental impacts, we selected the sample sites of similar saprobic status in slightly enriched zones (β-mesosaprobity), YIG, YCK and YSC. General descriptions of communities and ecological assessment of water quality on the Suyong River have been reported in Kwon and Chon (1993), Kang et al. (1995), and Yoon and Chon (1996).

For training with the network, the community data were organized according to the taxonomic hierarchy of Genus/Species, Family, Order, and Class and functional groups. For the convenience of handling data, as well as for alleviating problem of the difficulty in classification, Genus and Species were pooled to be the same hierarchical level in this study. The total number of Genus and Species used for training was 105, and that for Family, Order, Class, and functional feeding groups was 48, 19, 7, and 5, respectively. The data were pre-processed as previously mentioned.

Figs. 10.21a and 10.21b show examples of the actual field data used for training, Genus/Species (*X*) and Family (*Y*) respectively. The values of normalized density in each hierarchical group of benthic macroinvertebrates were expressed as different levels of contour lines. The name of each taxa is not listed since the list is too long and the names of taxa are not the issue in this chapter. Detailed ecological implementation of the counterpropagation will be referred to Park et al. (2001a). Fig. 10.21c shows trained results for Genus/Species (*X'*) matching to the input of Family data (*Y*). The overall conformation of community dynamics appeared to be similar between *X'* and *X*, confirming the input data were effectively extracted through the network. Generally the "averaging effect" appeared: dominant groups such as Chironomidae, Tipulidae (Diptera), and Hydropsychidae (Trichoptera) occurred consistently, while groups not frequently collected tended to disappear in the trained results (Fig. 10.21c). The reverse process for producing *Y'* (Fig. 10.20) was also possible and, in general, the trained results matched well with the actual data.

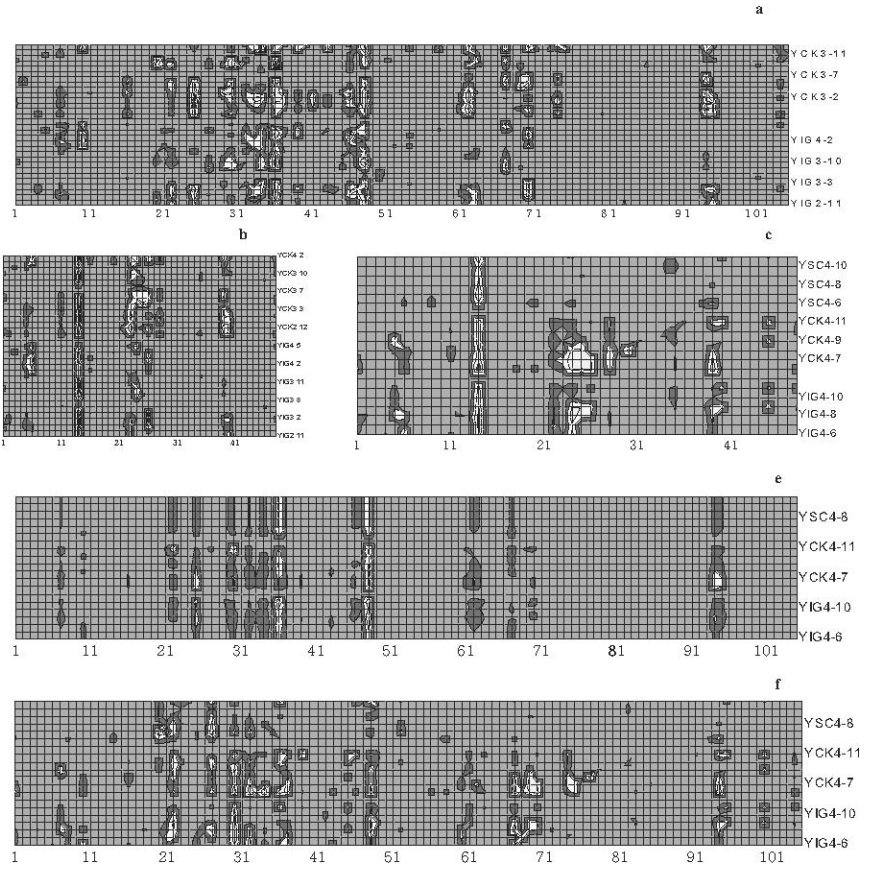


Fig. 10.21. Example of input data and test results in patterning benthic macroinvertebrates communities when traind with the counterpropagation network. a) and b) actual field data respectively for Genus/Species (X) and Family (Y), c) trained data for Genus/Species (X'), d) new field data for Family (Y_{new}), e) predicted data for Genus/Species with new field data (X''), and f) actual field data (X'''). (From Park et al. 2001a).

We further tried to test how the trained network correspondingly responded to new field data. Fig. 10.21d was another set of field data at the Family level, which had been collected at the study sites from June to November 1994, and have not been used for training. The new data set was given to the trained network, serving as Y_{new} as shown in Fig. 10.20. Fig. 10.21e shows the predicted results at Genus/Species level (X'') after new input data Y_{new} were provided to the trained network. Consequently this was comparable with the actual field data at

Genus/Species level, *X"""* (Fig. 10.21f). Similar to the case of *X"*, the overall conformational characteristics in densities were generally in accordance with the field data for Genus/Species. In comparing *X"* (Fig. 10.21e) and *X"""* (Fig. 10.21f), the "averaging effect" was also observed: groups rarely occurring tended to disappear while the dominant groups appeared more consistently. Patterning by the counterpropagation network is further described in Park et al. (2001a).

The results demonstrate the counterpropagation network could extract information on relationships among hierarchical levels in communities. Information regarding associations or relationships in communities would be valuable for verifying ecosystem functioning, and would assist greatly to interpret ecological status of the stream ecosystem. This type of patterning would especially useful for revealing inter-relationship among more than two groups at the same time. The patterning of multi-relational functioning in communities of benthic macroinvertebrates will be discussed in the future.

10.4.2 Patterning of Exergy

While community organization could be revealed through information of relationships among groups, another major aspect of ecological informatics on the other side is to develop an integrative expression of communities.

The integrative expression of ecological status of community, however, is difficult since community consists of many variables varying in a complex manner as mentioned previously. Although community develops progressively in one direction in general, it is difficult to simply represent the status of the community in one parameter; whether it is matured, disturbed or recovering for instance. However integrative diagnostics on community is essential for establishing sustainable management strategies in stream ecosystems.

In this regard, exergy could be a useful parameter to represent the overall status of community. Exergy is defined as the amount of work a system can perform when it is brought to thermodynamic equilibrium with its environment. Exergy could express the organization of the ecosystem by the living components, and represents the biomass of the system and the information that this biomass is carrying (Jørgensen 1992, 1994, 1995, 1997; Jørgensen et al. 1995).

It is possible, according to Jørgensen (1992, 1995), to calculate a relative exergy (*Ex*) contribution of biomass and information to an ecosystem as:

$$Ex = \sum_{i=1}^{n} (W_i C_i) \tag{10.17}$$

where C_i is the concentration (biomass in this case) of the *i*th state variable (i.e., selected taxa), W_i is the information stored in the *i*th state variable, and n is the number of variables.

In this study exergy was patterned by utilizing artificial neural networks (Park

et al. 2001b). Data for benthic macroinvertebrate communities (Fig. 10.22a) were provided for calculating exergy, and, according to Jørgensen (1997), values ranging from 29.6 – 43.9 were assigned as W_i for macroinvertebrates. In available information sources (Jørgensen 1997; Fonseca et al. 2000), however, only values of W_i at higher taxa are available, and the values for the order or family level in benthic macroinvertebrates are not specifically provided. Based on experiences from data analyses and field experiences we arbitrarily assigned 30 for Oligochaeta, Diptera, Chironomidae and Hirudinae, and 35 for Gastropoda, Ephemeroptera, Plecoptera, Trichoptera, Odonata, and Megoloptera in this study (Park et al., 2001b).

Training between Community and Exergy

By using artificial neural network, exergy could be predicted through the community data. The backpropagation algorithm (Rumelhart et al. 1986) was used for patterning the input (community) (Fig. 10.22a) and output (exergy) (Fig. 10.22b) data in a supervised manner. Benthic macroinvertebrate communities, collected in the streams monthly from Suyong River, in Korea, from October 1997 to September 1998, were used for field data (Fig. 10.22a). Community compositions reflecting the water quality in the Suyong River were reported in Kwon and Chon (1993), Kang et al. (1995) and Youn and Chon (1999). Fig. 10.22b shows monthly changes in the total exergy for each site in the Suyong and Soktae streams. The patterns of changes in exergy during the survey period occurred differently according to the sample locations. Although the sample sites were located in one river system and close to each other, the changes in exergy showed different patterns according to the sample sites' location and the level of pollution. Detailed discussion could the referred to Park et al. (2001b). Among data for community and exergy, about one third of the samples were set aside for testing, and the rest were used for training.

Training proceeds on an iterative gradient algorithm, and was similarly conducted based on the backpropagation algorithm as shown in the section 10.3.1. The number of nodes at the hidden layer used for this study was 5. The learning coefficient, which updates the weights at each iteration, was set to 0.7 in this study. The moment coefficient was set to 0.8 and the activation function coefficients were varied between 0.1 and 1.0. The convergence was generally reached in the iteration of 10,000 - 20,000 under the mean error term of 0.001. Trained data sets were accordingly matched to the original input data in the Soktae Stream (Fig. 10.23a) and in the Suyong Stream (Fig. 10.23b).

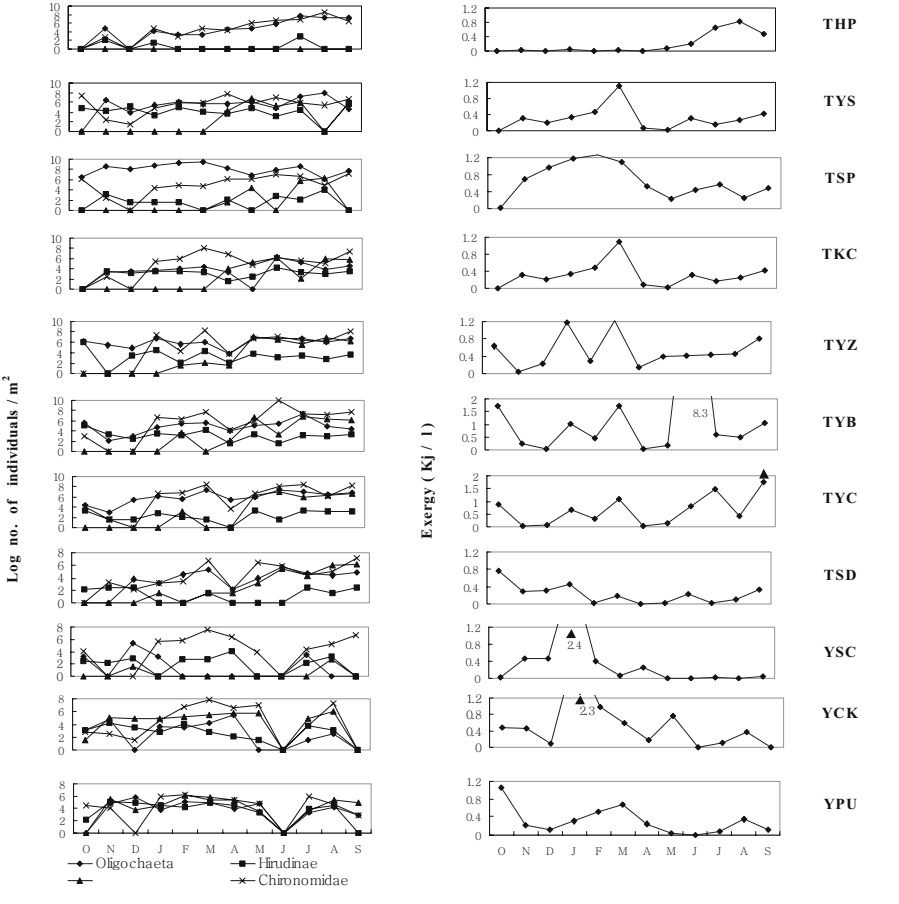


Fig. 10.22. Monthly changes in densities of benthic macroinvertebrate communities and exergy in the Soktae and Suyong streams in the Suyong River from October 1997 to September 1998. a) community, b) exergy. (From Park et al. 2001b).

The input and output data were also provided with time delays. Input data was the community for the previous time (*T*-1), while the exergy in the present time (*T*) would be output data in this case (Figs. 10.23a and 10.23b). The training conditions, such as error term and learning and momentum coefficients, were similar to the previous on-time training. Trained data sets were also accordingly matched to the original input data.

When new data were given to the trained network for testing, the network was able to predict exergy (Fig. 10.24). In general, it appeared that the predicted and actual field data were in accord, although some discrepancies were locally observed. In comparison with the trained results, however, the level of

coincidence between the field and predicted data tended to be low. The Pearson's correlation coefficients lay between the field and predicted data, ranging from 0.45 to 0.65 (F=2.64, df =24, P<0.01). It appeared that differences between the predicted and field data were more frequently observed in the time-delay training than in the on-time training. The occurrences of discrepancies were explainable in many case. They were usually due to the limited availability of data for training (Park et al., 2001b).

Patterning of Changes in Exergy

The self-organizing Kohonen network could be also applicable to patterning time development of exergies. Similar to the previous case, it is supposed to have a linear array of M^2 output neurons (i.e., computation nodes) in the Kohonen

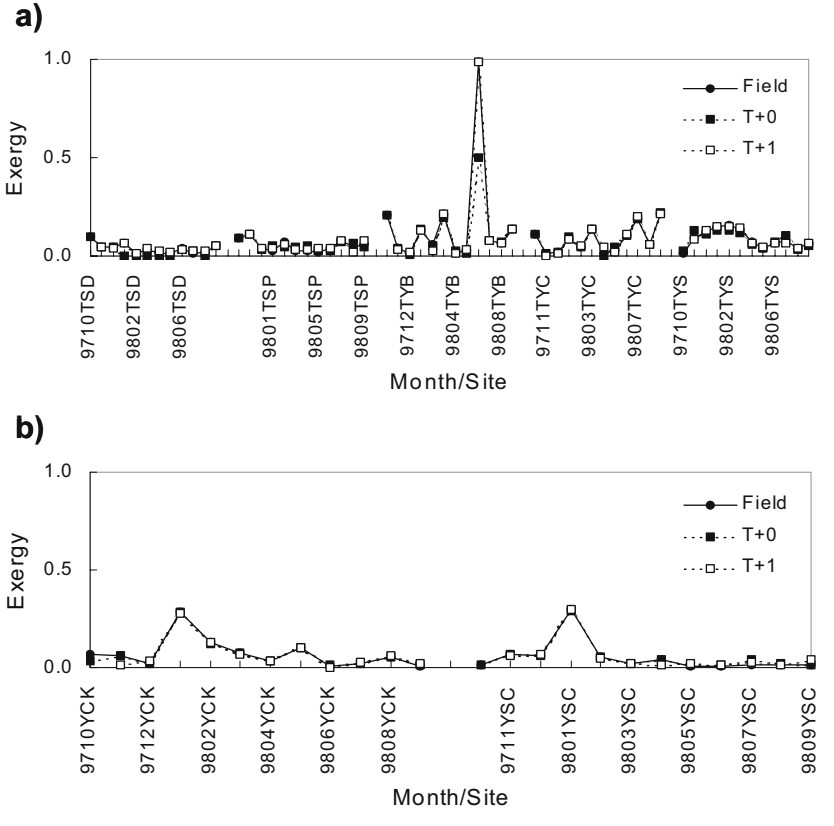


Fig. 10.23. Training by the backpropagation algorithm on exergy of selected taxa of benthic macroinvertebrate communities collected in the Suyong River from October 1997 to September 1998 "without time delay" and "with one-time delay". The four digit number and alphabets indicate year-and-month and sample sites respectively. a) Soktae Stream, b) Suyong Stream. (From Park et al. 2001b).

network, with each neuron being represented as *j* (See Fig. 10.3). The input vector ***x*** is considered to be an input layer to the Kohonen network, and the set of exergy measured from benthic macroinvertebrate communities for a certain period, e.g., 3 months, 4 months, etc., are provided as input to the network in this case. The training process was the same as for patternizing the community data in the section 10.1.1.1.

For patterning the changes in exergy, monthly measurements of exergy could be segmented in intervals in different periods, e.g., 3, 4, 6, etc. In each interval, the data sets for exergy of different taxa from the same sample sites were used as input to the Kohonen network for training. The exergy was normalized to the maximum of 3 kJ l^{-1} with the values ranging from 0.01 to 0.99. Detailed procedure could be referred to Park et al. (2001b). Learning rate was assigned at the values of 0.1 - 0.4.

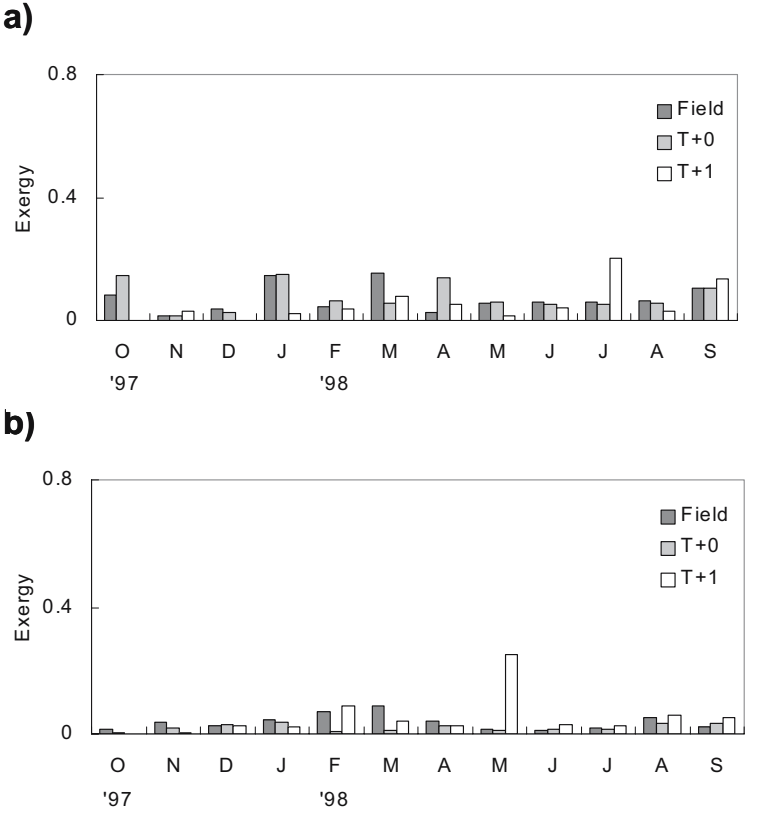


Fig. 10.24. Comparison with field and predicted data of exergy by the backpropagation algorithm on benthic macroinvertebrate communities collected in the Suyong River from October 1997 to September 1998 "without time delay" and "with one-time delay". a) Soktae Stream, b) Suyong Stream. (From Park et al. 2001b).

Fig. 10.25 shows the grouping of "changes in exergy" by the Kohonen network for different sample sites during the survey period. Since the number of samples was decreased as period of sampling time (e.g., 3 months, 4 months), the number of neurons in the Kohonen network was accordingly decreased as the period of training time was increased. In 3-month segments (8 × 8 neurons), some groupings occurred between "different sampling times" of the same sample sites or between "different sample sites" in similar sampling times (Fig. 10.25a). However, the mapping did not show any clear tendency. In 4-month segments (7 × 7 neurons), in contrast, groupings appeared in a clear pattern in the two basic types (Fig. 10.25b). Different sample sites were grouped according to the similar periods, or samples from different periods at the same sample site were patterned closely.

a)

	0	1	2	3	4	5	6	7
0	9804TKC 9805TKC 9809YCK		9802TSD 9802TYB 9802TYZ		9803TSD 9803TYB 9803TYZ		9804TSD 9804TYB 9804TYZ	
1	9801TKC		9809YPU		9805THP	9802TYC	9803TSP	
2	9802TKC 9803TKC				9801TSP 9805TYB		9804TSP	
3	9805YCK 9806YCK 9807YCK 9808YCK		9805YPU 9806YPU	9801THP	9805TYZ	9806TSP		9809TYZ
4		9803THP		9808TYC 9807TYZ 9808TYZ 9802YPU 9803YPU 9804YPU	9805TYC		9808TKC 9809TYB 9809TYC	9808THP 9807TYC
5							9806TYC	
6	9803YCK 9804YCK 9803YSC 9804YSC 9808YSC		9802THP	9809YSC	9808TYB 9807YPU		9807THP 9803TYC	
7	9807YSC						9804THP	9807TYS
8	9802YCK 9802YSC 9805YSC 9806YSC		9801TSD 9805TSD 9808TSD 9809TSP	9809TSD 9801YCK 9801YSC	9801TYB 9801TYC 9801TYZ 9808YPU	9806TKC 9802TSP 9805TSP	9801YPU	9808TYS

b)

	0	1	2	3	4	5	6
0	9809YPU	9808YSC	9809TYS	9803TYS 9804TYS 9805TYS 9806TYS 9807TYS 9808TYS		9807TYB 9807TYC 9807TYZ	9806TSP
1				9807THP			9809THP
2	9803YPU 9804YPU 9805YPU	9806YPU 9807YPU 9808YPU				9805TYC	
3				9808THP	9805THP	9805TYB 9805TYZ	
4			9804TYC				
5	9803THP				9807TKC 9807TSP		9805TKC 9806TKC
6	9809TYB 9809TYC 9809TYZ		9804TYB 9804TYZ		9803TSP		9803TKC 9804TKC
7	9809TSP						
8	9809TKC 9809TSD	9808TSD	9803TYB 9803TYC 9803TYZ	9804TSP		9803TSD 9804TSD 9805TSD 9806TSD	9807YSC

Fig. 10.25. Mapping of the benthic communities collected at the study sites in the Suyong River trained by the Kohonen network after training with exergy changes in four months. (The three-character alpha-codes strand for the name of the sample sites, and the four numerical digit appearing before the alpha-codes represent the month and year of collection. (i.e. April 1998 for 9804)). (From Park et al. 2001b).

Most examples of groupings in exergy changes by the trained Kohonen network had the corresponding characteristics of community dynamics in relation to environments. Examples of the first group were TSD, TYB and TYZ for the months of February, March and April 1998 (Listed months in the figure indicate the month at the end of the segment.), respectively representing the neurons for (2(*x* axis), 0(*y* axis)), (4,0) and (6,0) (Fig. 10.25b). In these periods, the densities of Oligochaetae and Chironomidae tended to increase in March and decrease in April consistently. Detailed ecological descriptions could be referred to Park et al. (2001b).

This study demonstrated that artificial neural networks could be useful for patterning changes in exergy. Although the sample sites were located in relatively close locations in the same river system, the changes in exergy appeared in diverse patterns. The Kohonen network was able to separate different patterns in the time development of exergy and demonstrated that the trends in exergy changes could be useful for characteristically explaining community development and environmental impacts (Park et al., 2001b).

10.5 Summary and Conclusions

Artificial neural networks were implemented to pattern and predict benthic macroinvertbrate community in streams. Properties of self-organization, adaptability and flexibility made of artificial neural networks were networks useful for extracting information out of complex community data in various ways: grouping for classification and ordination, prediction of community dynamics, verification of environmental impacts, and revealing organizational aspects of community.

Based on unsupervised learning with the Kohonen network and ART, groupings were efficiently conduced to classify and ordinate community data. The combined networks of ART and Kohonen were further utilized to group community changes. Short-time predictions of community dynamics were also possible through temporal application of artificial neural networks. The time-delayed multi-layer perceptron, and the partially and fully connected recurrent networks were able to forecast the future level of community abundance given by the previous data as input. The recurrent networks appeared to predict the temporal development of communities better. The fully connected recurrent network also effectively accommodated environmental factors, and the sensitivity analyses further revealed the impact of environmental factors on community dynamics.

The organizational informatics were also patterned by artificial neural networks. Patterns of relationships among different hierarchical levels in benthic macroinvertebrate communities were effectively elucidated by the counterpropagation network. The Kohonen network and multiplayer perceptron were further utilized to characterize exergy, an integrative parameter indicating thermodynamic information in community. Temporal exergy changes were

grouped by the Kohonen network and community-exergy relationships were effectively patterned by the multi-layer perceptron with the backpropagation algorithm.

Acknowledgements

The analysis in this research was in part supported by "KOSEF R001-2001-00087".

References

Allan JD (1995) Stream Ecology 'Structure and function of running waters'. Chapman & hall, 388 pp

Allen TFH, Starr TB (1982) Hierarchy. The University of Chicago Press, 310 pp

Boudjema G, Chau NP (1996) Revealing dynamics of ecological systems from natural recordings. Ecol. Model., 91, 15-23

Bunn SE, Edward DH, Loneragan NR (1986) Spatial and temporal variation in the macroinvertebrate fauna of streams of the northern jarrah forest, Western Australia: community structure. Freshwater Biology, 16, 67-91

Brosse S, Lek S, Townsend CR (2001) Abundance, diversity, and structure of freshwater invertebrates and fish communities: an artificial neural network approach. New Zealand Journal of Marine and Freshwater Research, 35, 135-145

Calow P, Petts GE (1994) The Rivers Handbook ' hydrological and ecological principles'. Blackwell Scientific Publications, 523 pp

Carpenter GA, Grossberg S (1987) ART2: self-organization of stable category recognition codes for analog input patterns. Applied Optics, 26, 4919-4930

Chon TS, Park YS, Moon KH, Cha EY (1996) Patternizing communities by using an artificial neural network. Ecol. Model., 90, 69-78

Chon TS, Kwak IS, Park YS (2000a) Pattern recognition of long-term ecological data in community changes by using artificial neural networks: Benthic macroinvertebrates and chironomids in a polluted stream. Korean J. Ecol., 23, 89-100

Chon TS, Park YS, Cha EY (2000b) Patterning of community changes in benthic macroinvertebrates collected from urbanized streams for the short time prediction by temporal artificial neural networks. In: Lek, S. and Guegan, J.F. (Eds.), Artificial Neuronal Networks: Application to Ecology and Evolution. Springer-Verlag, Berlin, pp. 99-114

Chon TS, Park YS, Park JH (2000c) Determining temporal pattern of community dynamics by using unsupervised learning algorithms. Ecol. Model., 132, 151-166

Chon TS, Kwak IS, Park YS, Kim TH, Kim YS (2001) Patterning and short-term predictions of benthic macroinvertebrate community dynamics by using a recurrent artificial neural network. Ecol. Model., 146. (In Press)

Cummins KW (1974) Structure and function of stream ecosystems. Bioscience, 24, 631-641

Cummins KW, Petersen RC, Howard FO, Wuycheck JC, Holt VI (1973) The utilization of leaf litter by stream detritovores. Ecology, 54, 336-345

Dimopoulos Y, Bourret P, Lek S (1995) Use of some sensitivity criteria for choosing networks with good generalization ability. Neural Processing Letters, 2,1-4

Elizondo DA, McClendon RW, Hoongenboom G (1994) Neural network models for predicting flowering and physiological maturity of soybean. Transactions of the ASAE, 37, 981-988

Elman JL (1990) Finding structure in time. Cognitive Science, 14, 179-211

Fonseca JC, Marques JC, Paiva AA, Freitas AM, Madeira VMC, Jørgensen SE (2000) Nuclear DNA in the determination of weighing factors to estimate exergy from organisms biomass. Ecol. Model., 126, 179-189

Foody GM (1999) Applications of the self-organising feature map neural network in community data analysis. Ecol. Model., 120, 97-107

Giles CL, Kuhn GM, Williams RJ (1994) Dynamic recurrent neural networks: theory and applications. IEEE Transactions on Neural Networks, 5 153-156

Giraudel JL, Aurelle D, Berrebi P, Lek S (2000) Application of the self-organising mapping and fuzzy clustering to microsatellite data: How to detect genetic structure in brown trout (*Salmo trutta*) populations. In: Lek, S. and Guegan, J.F. (Eds.), Artificial Neuronal Networks: Application to Ecology and Evolution. Springer-Verlag, Berlin, pp. 187-202

Grossberg S (1969) On the production and release of chemical transmitters and related topics in the cellular control. J. Theor. Biol., 22, 325-364

Grossberg S (1982) Studies of Mind and Brain: Neural Principals of Learning, Perception, Development, Cognition, and Motor Control. Reidel Press, Boston

Hauer FR, Lamberti GA (1996) Methods in Stream Ecology. Academic Press, 674 pp

Hawkes HA (1979) Invertebrates as indicators of river water quality. In: James, A. and Evision, L. (Eds.), Biological indicators of water quality. John Wiley and Sons, Chishester, Great Britain, pp. 2.1-2.45

Haykin S (1994) Neural Networks. Macmillian College Publishing Company, 696 pp

Hecht-Nielsen R (1987) Counter propagation networks. Proc. of the Int. Conf. On Neural networks, II, 19-32, IEEE Press, New York, June 1987

Hecht-Nielsen R (1990) Neurocomputing. Addison-Wesley, New York, 433 pp

Hellawell JM (1986) Biological indicators of freshwater pollution and environmental management. Elsevier, London, 546 pp.

Hopfield JJ (1982) Neural Networks and Physical Systems with Emergent Collective Computational Abilities. Proc. Natl. Acad. Sci. USA, Vol. 79, 2554-2558, April

Huntingford C, Cox PM (1996) Use of statistical and neural network techniques to detect how stomatal conductance responds to changes in the local environment. Ecol. Model., 97, 217-246

Hynes HBN (1960) The biology of polluted waters. Liverpool Univ. Press. London, 202 pp

Jørgensen SE (1992) Parameters, ecological constraints and exergy. Ecol. Model., 62, 163-170

Jørgensen SE (1994) Review and comparison of goal functions in system ecology. WIE MILIEU, 44, 11-20

Jørgensen SE (1995) Exergy and ecological buffer capacities as measures of ecosystem health. Ecosys. Health, 1, 150-160

Jørgensen SE (1997) Integration of ecosystem theories: A pattern, 2nd edition. Kluwer, Dordrecht, 400 pp

Jørgensen SE, Nielsen SN, Mejer HF (1995) Emergy, environ exergy and ecological modeling. Ecol. Model., 77, 99-109

Kamgar-Parsi B, Gualtieri JA, Devancy JE, Kamgar-Parsi B (1990) Clustering with neural networks. Biol. Cybern., 63, 201-208

Kang DH, Chon TS, Park YS (1995) Monthly changes in benthic macroinvertebrate communities in different saprobities in the Suyong and Soktae streams of the Suyong river. Korean J. Ecol., 18, 157-177

Kohonen T (1989) Self-organization and associative memory. Springer-Verlag, Berlin, 312 pp

Kung SY (1993) Digital Neural Networks. Prentice Hall, Englewood Cliffs, New Jersey, 444 pp

Kwon TS, Chon TS (1993) Ecological studies on benthic macroinvertebrates in the Suyong River. III. Water quality estimations using chemical and biological indices. Kor. J. Limnol., 26, 105-128

Legendre P, Legendre L (1987) Developments in numerical ecology. Springer-Verlag, Berlin 585 pp

Legendre P (1987) Constrained clustering. In: Legendre, P. and Legendre, L. (Eds.), Developments in numerical ecology. Springer-Verlag, Berlin. Germany, 289-307 pp

Legendre P, Dallot S, Legendre L (1985) Sucession of species within a community: chronological clustering, with applications to marine and freshwater zooplankton. Am. Nat., 125, 257-288

Lek S, Guegan JF (1999) Artificial neural networks as a tool in ecological modelling, an introduction. Ecol. Model., 120, 65-73

Lek S, Guegan JF (2000) Neuronal Networks: Algorithms and Architectures for Ecologists and Evolutionary Ecologists. In: Lek, S. and Guegan, J.F. (Eds.), Artificial Neuronal Networks: Application to Ecology and Evolution. Springer-Verlag, Berlin, pp. 3-27

Lek S, Delacoste M, Baran P, Dimopoulos I, Lauga J, Aulagnier S (1996) Application of neural networks to modelling nonlinear relationships in ecology. Ecol. Model., 90, 39-52

Levine ER, Kimes DS, Sigillito VG (1996) Classifying soil structure using neural networks. Ecol. Model., 92, 101-108

Lippmann RP (1987) An Introduction to computing with neural nets. IEEE ASSP Magazine, April. pp. 4-22

Lohninger H, Stanc F (1992) Comparing the performance of neural networks to well-established methods of multivariate data analysis: the classification of mass spectral data. Fresenius J. Anal. Chem., 344, 186-189

Ludwig JA, Reynolds JF (1988) Statistical ecology: a primer on methods and computing. John Wiley and Sons, New York, 329 pp

McCulloch WS, Pitts W (1943) A logical calculus of the ideas imminent in nervous activity, Bulletin of Mathematical Biophysics, 5, 115-133

Melssen WJ, Smits JRM, Rolf GH, Kateman G (1993) Two-dimensional mapping of IR spectra using a parallel implemented self-organising feature map. Chemometrics and Intelligent Laboratory Systems, 18, 195-204

Norusis MJ (1986) SPSS/PC+ advanced statistics. SPSS inc., Chicago

O'Neill RN, DeAngelis DL, Waide JB, Allen TFH (1986) A hierarchical concept of ecosystems. Princeton University Press, Princeton, 253 pp

Pao YH (1989) Adaptive pattern recognition and neural networks. Addison-Wesley Publishing Company, Inc., New York, 309 pp

Park YS, Kwak IS, Cha EY, Lek S, Chon TS (2001a) Relational patterning on different hierarchical levels in communities of benthic macroinvertebrates in an urbanized stream using an artificial neural network. J. Asia-Pacific Entomol. (Submitted)

Park YS, Kwak IS, Chon TS, Kim JK, Jørgensen SE (2001b) Implementation of artificial neural networks in patterning and prediction of exergy in response to temporal dynamics of benthic macroinvertebrate communities in streams. Ecol. Model., 146. (In Press)

Quinn MA, Halbert SE, Williams III L (1991) Spatial and temporal changes in aphid (Homoptera: Aphididae) species assemblages collected with suction traps in Idaho. J. Econ. Entomol., 84, 1710-1716

Recknagel F, French M, Harkonen P, Yabunaka KI (1997) Artificial neural network approach for modelling and prediction of algal blooms. Eco. Model., 96, 11-28

Recknagel F, Wilson H (2000) Elucidation and prediction of aquatic ecosystems by artificial neuronal networks, In: Lek, S. and Guegan, J.F. (Eds.), Artificial Neuronal Networks: Application to Ecology and Evolution. Springer-Verlag, Berlin, pp. 143-155

Rumelhart DE, Hinton GE, Williams RJ (1986) Learning internal representations by error propagation. In: Rumelhart, D.E. and McCelland, J.L. (Eds.), Parallel distributed processing: Explorations in the microstructure of cognition, Vol. I: Foundations, MIT Press, Cambridge, pp. 318-362

Salvador R, Piñol J, Tarantola S, Pla E (2001) Global sensitivity analysis and scale effects of a fire propagation model used over Mediterranean shrublands. Ecol. Model., 136, 175-189

Scardi M (2000) Neuronal network models of phytoplankton primary production, In: Lek. S. and Guegan, J.F. (Eds.), Artificial Neuronal Networks: Application to Ecology and Evolution. Springer-Verlag, Berlin, pp. 116-129

Sladecek V (1979) Continental systems for the assessment of river water quality. In: James, A. and Evison, L. (Eds.), Biological Indicators of Water Quality. John Wiley & Sons, Chichester, pp. 3.1-3.32

Spellerberg IF (1991) Monitoring Ecological Change. Cambridge University Press, 334 pp

Stankovski V, Debeljak M, Bratko I, Adamic M (1998) Modelling the population dynamics of red deer (*Cervus elaphus* L.) with regard to forest development. Ecol. Model., 108, 143-153

Tan SS, Smeins FE (1996) Predicting grassland community changes with an artificial neural network model. Ecol. Model., 84, 91-97

Tittizer TT, Koth P (1979) Possibilities and limitations of biological methods of water analysis. In: James, A. and Evison, L. (Eds.), Biological Indicators of Water Quality. John Wiley and Sons, Chichester, Great Britain, pp. 4.1-4.21

Tuma A, Haasis HD, Rentz O (1996) A comparison of fuzzy expert systems, neural networks and neuro-fuzzy approaches controlling energy and material flows. Ecol. Model., 85, 93-98

Wasserman PD (1989) Neural computing: Theory and practice. Van Nostrand Reinhold, New York

Welch EB, Lindel T (1992) Ecological effects of wastewater 'Applied limnology and pollutant effects'. Chapman & Hall, 425 pp

Williams RJ, Zipser D (1989) A learning algorithm for continually running fully recurrent neural networks. Neural Computation, 1, 270-280

Wray J, Green GGR (1994) Calculation of the Volterra kernels of non-linear dynamic systems using an artificial neural network. Biol. Cybern., 71, 187-195

Yoon BJ, Chon TS (1996) Community analysis in chironomids and biological assessment of water qualities in the Suyong and Soktae streams of the Suyong River. Kor. J. Limnol., 29(4), 275-289

Zar JH (1984) Biostatistical Analysis. Prentice-Hall International, Inc, New Jersey, 718 pp

Zurada JM (1992) Introduction to artificial neural systems. West Publishing Company. New York, 683 pp

Chapter 11

Elucidation of Hypothetical Relationships between Habitat Conditions and Macroinvertebrate Assemblages in Freshwater Streams by Artificial Neural Networks

H. Hoang · F. Recknagel · J. Marshall · S. Choy

11.1 Introduction

It has been widely demonstrated that interactions among chemical and physical processes create environmental conditions at a range of scales that strongly influence the distribution and abundance of lotic biota, and thus the composition of macroinvertebrate assemblages (e.g. Hynes 1970). Many studies have identified substrate composition, complexity and heterogeneity as major determinants of in-stream biota (e.g. Downes et al. 1998). Other abiotic factors such as flow velocity (e.g. Barmuta 1990) and water chemistry (e.g. Bunn et al. 1986) have also been found to influence biotic composition.

The insight that local species assemblages are a reflection of local environmental conditions is fundamental to biomonitoring programmes increasingly incorporated into water resource management practices throughout the world. Statistical models have been developed to predict the occurrence of macroinvertebrate taxa based on their association with environmental variables (e.g. Wright 1995; Reynoldson et al. 1997; Simpson et al. 1997). Machine learning techniques, such as artificial neural networks (ANN), have recently been applied to this problem and show promise to provide greater predictive capacity than statistical modelling techniques (Chon et al. 1996; Walley and Fontama 1998; Pudmenzky et al. 1998; Schleiter et al. 1999).

In the context of this chapter a series of ANN models were developed based on both the 'clean water' (Huong et al. 2001) and the 'dirty water' approach (Huong 2001) which accurately predicted the presence and absence of most common macroinvertebrate taxa in the stream system of Queensland, Australia. The referential 'clean water' approach (Reynoldson et al. 1997) aimed at the prediction of fauna at impacted sites assuming they were unimpacted. The 'dirty water'

approach was used to identify input variables that exert some influence on outputs and to predict the ecological consequences of altering input variables by simulating various scenarios. The latter approach utilised a broader range of input variables and data of the Queensland stream system including sites that were affected by anthropogenic impacts.

Sensitivity analyses were conducted for each single ANN model in order to refine the selection of input variables and strengthen the models' validity. However the graphical representations of sensitivity results also illustrated the nature of relationships between environmental variables and the occurrence of macroinvertebrate taxa. Selected results of the sensitivity analysis from 'clean water' and 'dirty water' models of the Queensland stream system are documented in this chapter and findings are discussed in the context of literature knowledge on stream macroinvertebrates.

11.2 Study Sites

The Queensland river and stream network spreads over the territory of the federal state of Queensland (Australia). The climate conditions of Queensland range from high rainfall areas (1600 mm/annum) in the tropical Northeast to low rainfall areas (200 mm/annum) in the Southeast. Study sites are representative for the catchments of all major and minor rivers.

11.3 Materials and Methods

11.3.1 Data

A comprehensive database of the Queensland stream system was used for the development of the neural network models. The database was divided into 897 datasets of reference sites and 1159 datasets of test sites.

Each site-specific dataset contained 39 physical variables, 17 potentially impacted environmental variables and colonisation patterns of 40 macroinvertebrates taxa at family level. Different combinations of data were used for the development of specific models.

11.3.2
Neural Network Modelling

The 'clean water' ANN models (Hoang et al. 2001) were developed by using data from so-called 'reference sites' of the Queensland stream system that were considered to be minimally affected by anthropogenic disturbance (see Conrick and Cockayne 2000). Therefore only those environmental variables were chosen as model inputs considered being relatively stable under the influences of human impacts.

Data used for the development of the 'dirty water' ANN models were taken from both reference and degraded sites containing physical and chemical variables.

Predictive ANN models were developed for each macroinvertebrate taxa (mostly families) recorded in the Queensland streams database. ANN training was carried out by the feed-forward back-propagation algorithm (Rumelhart, Hinton and Williams 1986) and the sigmoid transfer function (see Fig. 11.1). The models were validated regarding their correct predictions of macroinvertebrate occurence either for reference sites ('clean water') or for reference and impacted sites ('dirty water'). The 'clean water' models achieved an average prediction accuracy of 82% (Hoang et al. 2001). The 'dirty water' models achieved an average prediction accuracy of 97% (Hoang 2001). Validation results of both approaches are summarized in Fig. 11.2.

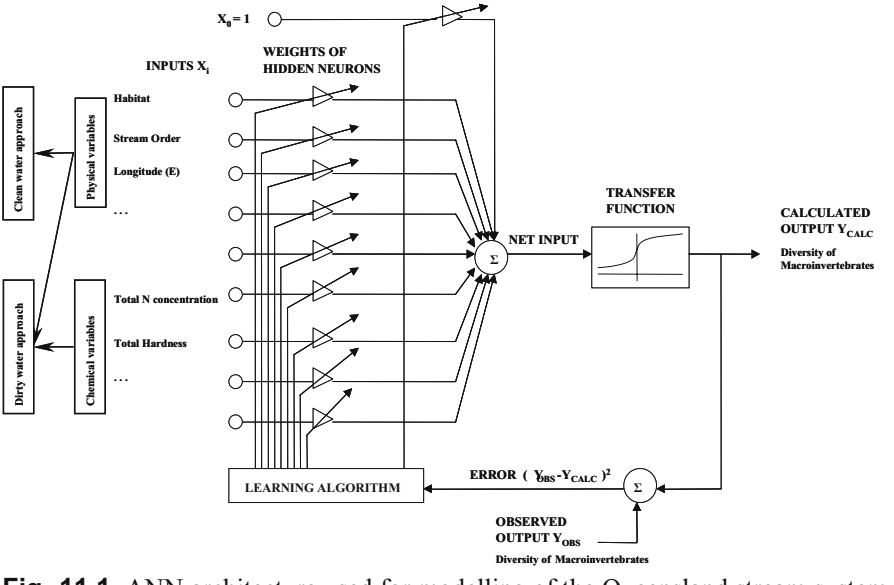


Fig. 11.1. ANN architecture used for modelling of the Queensland stream system considering both the 'clean water' and 'dirty water' approach

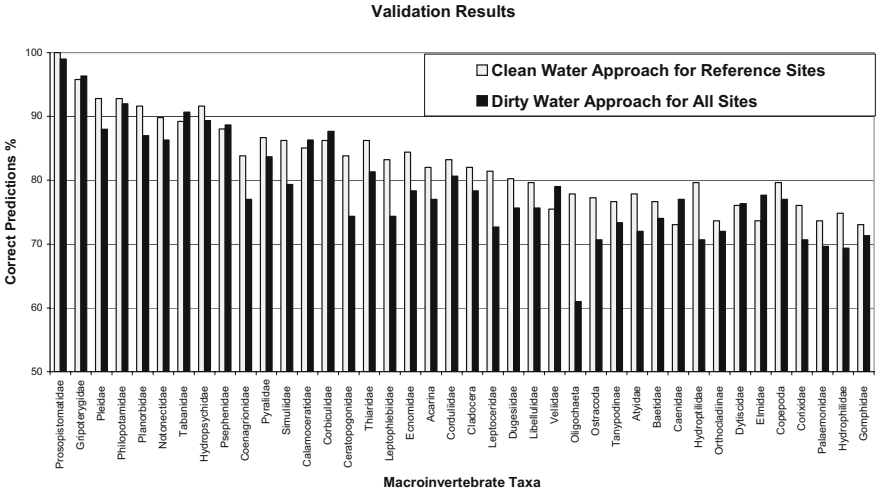


Fig. 11.2. Validation results achieved by ANN modeling of the Queensland stream system based on both the 'clean water' and 'dirty water' approach

11.3.3 Sensitivity Analysis

A comprehensive sensitivity analysis was conducted for all 'clean water' and 'dirty water' ANN models of macroinvertebrate taxa. Each input variable was varied within the range of its mean +/- five standard deviations while the remaining inputs were kept at their respective means. The model outputs were computed for 150 steps above and below the mean, with each step therefore equivalent to one thirtieth of a standard deviation. Resulting graphs of the input-output relationships over the range of the varied inputs illustrated how the varying environmental parameters influenced the predicted probability of macroinvertebrate occurrences.

Even though in the first instance the sensitivity analysis was used to improve the ANN models' validity by selecting the most sensitive input variables, it also demonstrated its potential to provide invaluable insights into the nature of relationships between the streams' environmental conditions and occurrence of macroinvertebrates.

11.4
Results and Discussion

As detailed below selected results of input-output relationships discovered by the sensitivity analyses are discussed that are either complementary or contradictory to existing theories on the ecology of stream macroinvertebrates.

11.4.1
Elucidation of Hypothetical Relationships

Cladocera (water flea) is known to be actively swimming zooplankton that prefers slow flowing water at reasonable depth. Such conditions would typically occur in streams that tend towards high order down land streams at low altitudes with large channel widths when increasing water depth would correspond with a slightly decreasing flow velocity. The sensitivity curve in Figure 11.3c clearly indicates that the upper range of depths from 0.2 to 1.2 m favors *Cladocera.*

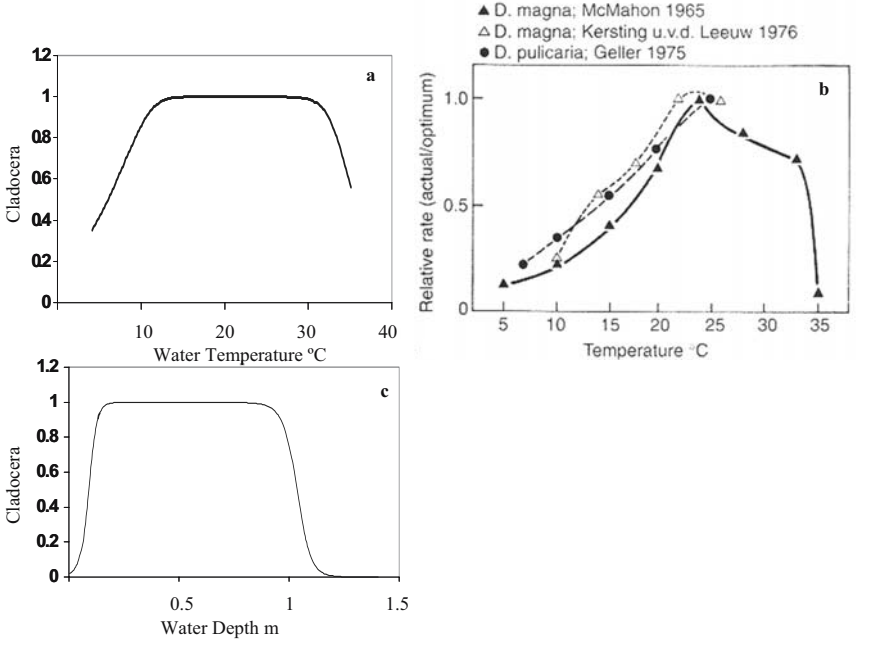


Figure 11.3. Relationships of *Cladocera* with a) water temperature and c) water depth as discovered by sensitivity analysis. b) Physiological activity of Daphnia in relation to water temperature as summarized by Lampert and Sommer (1997)

Figure 11.3b represent unimodal curves on optimal temperature conditions for physiological processes such as ingestion and reproduction rates of several *Daphnia* species extracted from laboratory experiments as summarized in Lampert and Sommer (1997). It shows that a decrease in physiological activity rates above the maximum is usually more rapid than the increase in the rates at sub-optimal temperature. A similar shaped sensitivity curve was discovered for the relationship between *Cladocera* occurrence and water temperature (Figure 11.3a) that indicated a similar optimum temperature range from 10 to 30ºC as in Fig. 11.3b.

Baetidae are known to be common in clear, cold streams (Suter 1996). They belong to the mayflies that emerge first in spring and may occur on warm days in late winter. In Queensland, *Baetidae* was observed in southern parts but never found in tropical areas. Hawking & Smith (1997) characterise *Baetidae* as fast swimmer where their nymphs prefer deep habitat. The sensitivity curves in Fig. 11.4 indicated these relationships very well.

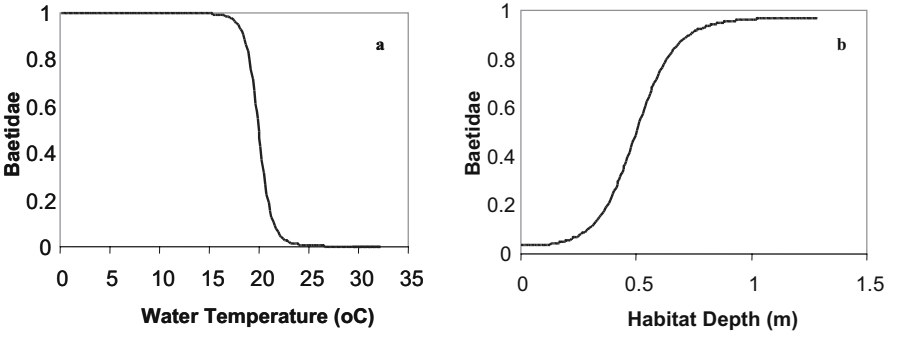


Figure 11.4. Relationships of *Baetidae* with a) water temperature and b) water depth as discovered by sensitivity analysis.

Chironomids (midge larvae) tend to be highly abundant in freshwater ecosystem such as streams. The two subfamilies *Tanypodinae* and *Orthocladiinae* were monitored in Queensland streams. *Orthocladiinae* are known for their cold-stenothermic nature that makes them abundant in subalpine and mountain streams, where maximum water temperatures in summer reach 10°C. In middle and lowland streams, where water temperature may exceed 20°C, the abundance of *Orthocladiinae* decreases significantly (Lindegaard and Brodersen 1995). The sensitivity curves in Fig. 11.5 indicated the preferred occurrence of *Orthocladiinae* at upper stream reaches (Fig. 11.5a) and cold water (Fig. 11.5b).

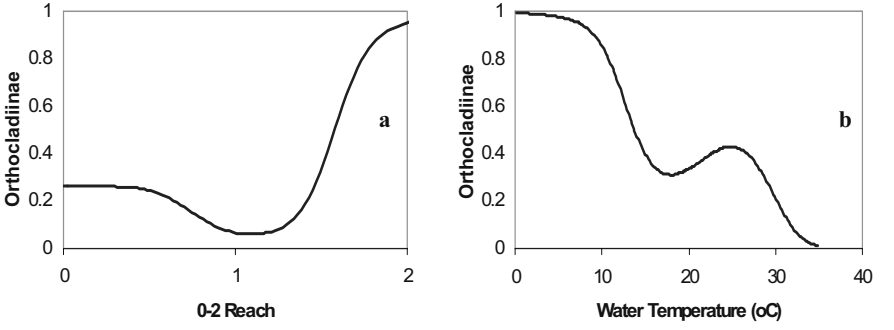


Fig. 11.5. Relationships of *Orthocladiinae* occurrence with a) stream reaches and b) water temperature as discovered by sensitivity analysis.

Tanypodinae occur very rarely in montane and sub alpine streams but become more abundant in low-order downstreams with increasing water temperature (Lindegaard and Brodersen 1995). The same trend was discovered by the sensitivity curves in Fig. 11.6.

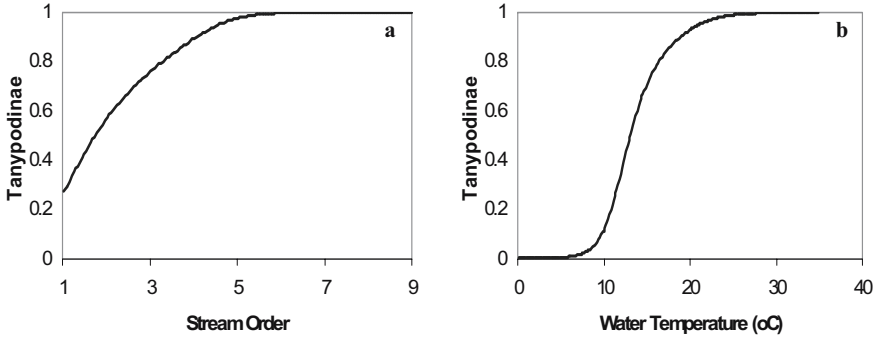


Fig. 11.6. Relationships of *Tanypodinae* occurrence with a) stream order and b) water temperature as discovered by sensitivity analysis.

Planorbidae is a cosmopolitan family of mainly left-coiled freshwater snails, that prefer low salinity, and feed on algae or waterweed. Some species feed on dead leaves or other debris in slow-flowing rivers. *Planorbidae* were observed in highly polluted, oxygen poor and/or very deep habitats. They can cope with such extreme conditions by utilising haemoglobin or other respiratory modifications. Some are known to exhibit considerable drought resistance. *Planorbidae* often are the dominant molluscs at a site (Smith 1996). They have been observed in oxygen poor and/or very deep habitats and rarely occur under conditions of high slope, where water is well aerated The sensitivity curve in Fig. 11.7a indicated that

Planobidae can survive under drought conditions with even extremely low dry-season-monthly-mean (DSMM) rainfall. Fig. 11.7b indicates *Planobidae*'s preference of low slope sites.

Norton et al. (1988) suggested that *Acarina* (water mites) are demographically at least as conservative as soil-dwelling relatives. They live in cold, oligotrophic waters and have multi-year generation times.

Sensitivity curves in Fig. 11.8 revealed that *Acarina* occurred preferentially at low temperatures in Queensland streams (Fig. 11.8a), and low nitrogen (Fig. 11.8b) and phosphorus concentrations indicating oligotrophic conditions.

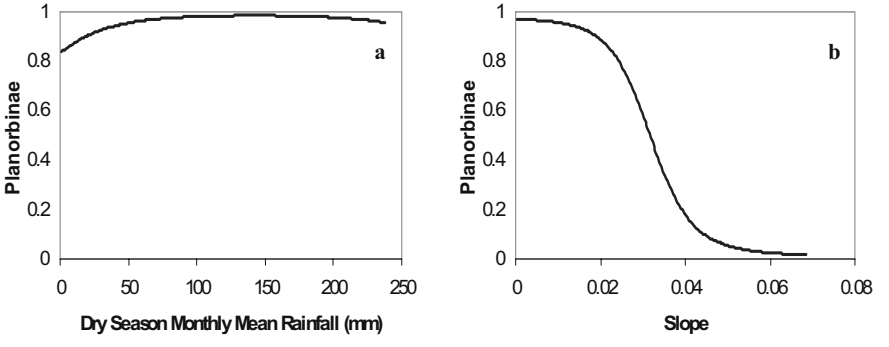


Fig. 11.7. Relationships of *Planobidae* occurrence with a) dry-season-monthly-mean (DSMM) rainfall and b) stream slope as discovered by sensitivity analysis.

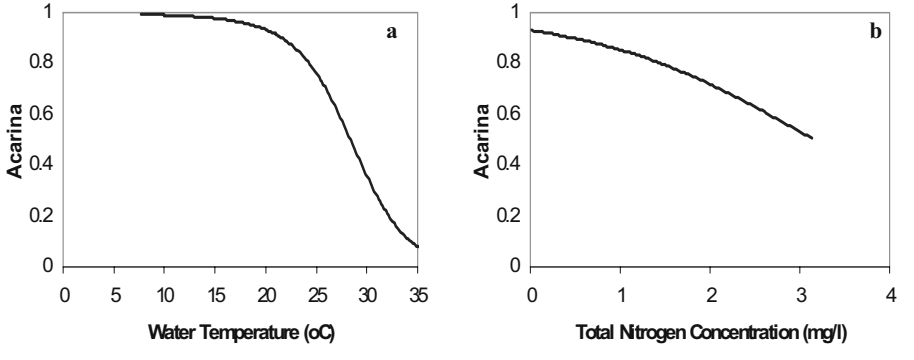


Fig. 11.8. Relationships of *Acarina* occurrence with a) water temperature and b) nitrogen concentrations as discovered by sensitivity analysis.

11.4.2
Discovery of Contradictory Relationships

Even though most sensitivity results revealed relationships complementary to literature findings, only some results where contradictory as shown in Fig. 11.9.

Giller and Malmqvist (1998) observed that the majority of *Dugesiidae* (triclads) that live in streams are cold-living species. However the sensitivity curve in Fig. 11.9a shows for Queensland streams that *Dugesiidae* were only present at water temperatures of more than 20°C.

Libellulidae are of tropical origin but the sensitivity curve in Fig. 11.9b indicates their absence at latitudes above –20(S) in Queensland streams, which characterise tropical zones of Queensland.

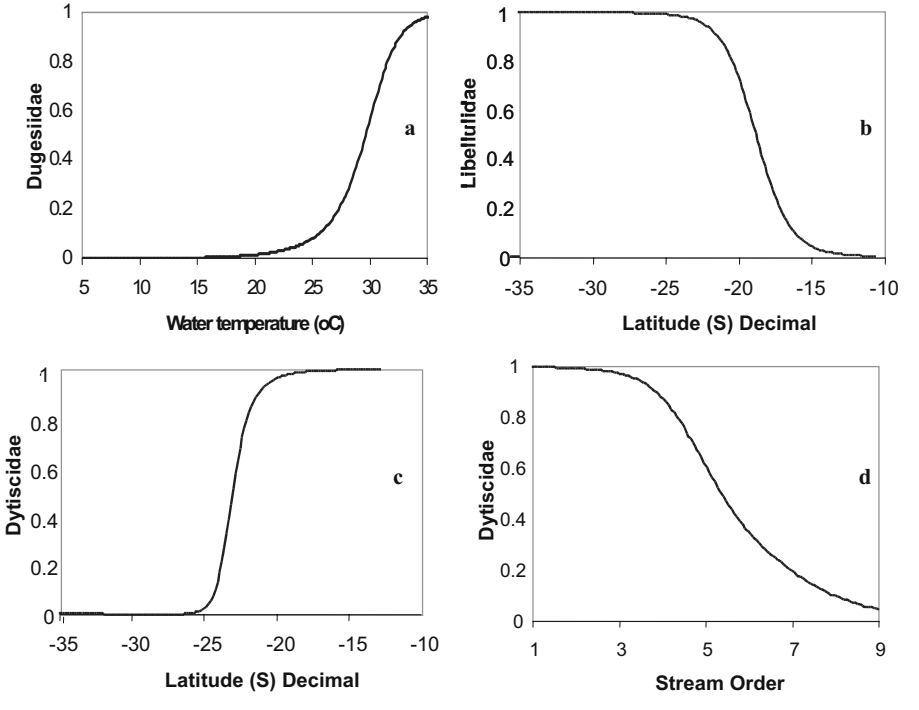


Fig. 11.9. Relationships of a) *Dugesiidae* occurrence with water temperature, b) *Libellulidae* occurrence with latitude, c) *Dytiscidae* occurrence with latitude and d) *Dytiscidae* occurrence with stream order as discovered by sensitivity analysis

The occurrence of *Dytiscidae* is suggested to be highly abundant in the southeastern parts of Australia and most common in littoral areas (Lawrence and Britton 1991). However the sensitivity curves in Figs. 11c and d showed that they were only observed in North Queensland at low-order streams. The findings from these sensitivity results, which were apparently contradictory to previous knowledge may suggest further research in this area.

11.4.3 Limitations of the Method

The present sensitivity analysis investigated only relationships between one single input and a specific output. However the occurrence of macroinvertebrates is always the result of multivariate nonlinear patterns of habitat conditions. Most habitat parameters do not occur in isolation but are closely interrelated.

The interrelationship between stream current, water temperature and oxygen demand of macroinvertebrates is an example. The current continually replenishes water and hence also oxygen in the immediate vicinity of the respiratory surfaces of the animals, and quite low levels can be tolerated in strong currents that renew oxygen at a high rate. Generally, metabolic rates and oxygen demand are higher of stream invertebrates than of still water forms at a given temperature. Respiration is temperature-related and rates can increase by 10% or more per 1ºC temperature rise. Thus increased temperature does not only reduce oxygen availability but it also increases oxygen demand that can add to the physiological stress of organisms (Giller and Malmqvist 1998).

The most important hydraulic characteristic for individual organism is the prevailing current velocity striking the organism head-on (Statzner et al. 1988). Macroinvertebrate species react differently to current velocity and show differential preferences. As a consequence different flow conditions lead to divergent assemblages of organisms. In a detailed survey by Quinn and Hickey (1994), boundary layer Re (Renolds number) was the most strongly correlated individual variable with invertebrate distribution and taxa richness in two New Zealand streams. However a combination of mean velocity, substrate size, and depth gave stronger correlation than any single variable. It appears that the interaction between current velocity and stream substrate size is particularly important in determining invertebrate distributions.

Orth and Maughan (1983) identified optimum velocity, depth, and substrate as determining factors for major taxa of benthic macroinvertebrates of warm-water woodland stream. The combination of current velocity of 60cm/sec, a depth of 34 cm and rubble-boulder substrate resulted in optimal diversity of benthic assemblages. Taking into account that habitat selection by benthos may be based on factor combinations, the investigators derived “joint preference factors” using the product of the individual preference factors.

Another example for the fact that ecological patterns underlie multivariate factors is indicated by the relationship between mean species richness and pH (Hildrew and Townsend 1987). Streams with a pH as high as 6.5 but low alkalinity (low Ca2+) often show similar features as acidic waters with pH< 5.5 (Willoughby and Mappin 1988). Effects of pH on aquatic fauna are different at different water temperatures (Hynes 1970). Food supply also depends on current speed, either to convey particles to filter feeding organisms or to deposit detritus (Hellawell 1986). Toxicity of ammonia and hydrogen sulfide to aquatic organism is dependent on both temperature and pH conditions.

These and other examples illustrate the multivariate effects of different habitat conditions on distributions of macroinvertebrates. To study the effect of individual variable while keeping all other variables at their respective means ignores this fact. Further research needs to consider techniques for multivariate sensitivity analysis in order to elucidate aquatic habitat conditions.

11.5 Conclusions

The sensitivity analyses by means of validated ANN models can contribute to improved understanding of the ecology of streams and rivers. The interpretation of resulting sensitivity curves may reveal impacts of environmental conditions on the occurrence of macroinvertebrate taxa. Such additional knowledge can be useful for the bioindication of stream habitats by means of macroinvertebrate assemblages, and enhance our capacity to monitor and mitigate stream ecosystems. The shape of the sensitivity curves of taxa would indicate how important it is to manage disturbances within certain bounds in order to maintain healthy aquatic ecosystems. Taxa with a threshold response to a disturbance appear to be eliminated at a stream site that proves to be beyond a certain disturbance level. Taxa with ramp responses would gradually become rarer as disturbance intensified. The identification of such threshold conditions would provide catchment and water resource managers with a powerful tool.

Overall it can be concluded that ANN provide a powerful tool for stream modelling allowing the user not only to achieve highly accurate predictions but discover information on general trends in the data. Therefore, this methodology can efficiently be applied to determine ecological requirements of stream organisms that are not fully understood.

References

Barmuta LA (1990) Interaction between the effects of substratum, velocity and location on stream benthos: an experiment. Australian Journal of Marine and Freshwater Research, 41, 557-573

Bunn SE, Edward DH, NR Loneragan (1986) Spatial and temporal variation in the macroinvertebrate fauna of streams of the northern jarrah forest, Western Australia. Freshwater Biology, 16, 67-91

Chon TS, Park YS, Moon KH, Cha EY (1996) Patternizing communities by using an artificial neural network. Ecol. Modelling 90, 69-78

Conrick DL, Cockayne BJ (Eds) (2000) Biological Monitoring and Assessment of Freshwaters using Macroinvertebrates. Background Information, Sampling and Analytical Procedures. Queensland Department of Natural Resources, Brisbane

Downes BJ, Lake PS, Glaister A, Webb JA (1998) Scales and frequencies of disturbance: rock size, bed packing and variation among upland streams. Freshwater Biology 40, 625-639

Giller PS, Malmqvist B (1998) The Biology of Streams and Rivers. Oxford, NewYork, Oxford University Press

Hawking JH, Smith FJ (1997) Colour Guide to Invertevrates of Australian Inland Water. Albury, NSW. CRC for Freshwater Ecology Identification Guide No.8

Hellawell JM (1986) Biological indicators of freshwater pollution and environmental management. London & NewYork. Elsevier

Hildrew AG, Townsend CR (1987) Organization in freshwater benthic communities. In: Organization of communities: past and present. Eds J.H.R. Gee and P.S. Giller. Oxford, Blackwell

Hoang H, Recknagel F, Marshall J, Choy S (2001) Predictive Modelling of Macroinvertebrate Assemblages for Stream Habitat Assessments in Queensland (Australia). Ecological Modelling 146, 1-3, 195-206

Hoang H (2001) Predicting Freshwater Habitat Conditions by the Distribution of Maroinvertebrates Using Artificial Neural Network. Thesis submitted for Master of Applied Science. Adelaide University, Adelaide

Hynes HBN (1970) The Ecology of Running Waters. Liverpool University Press, Liverpool, p 555

Lampert W, Sommer U (1997) Limnoecology: The Ecology of Lakes and Streams. NewYork, Oxford University Press

Lawrence JF, Britton EB (1991) Coleoptera. Insects of Australia. CSIRO. Carlton, Victoria. Melbourne University press. 2: 543-683

Lindegaard C, Brodersen KP (1995) Distribution of Chironomidae in the river continuum. In: Chironomids: from Genes to Ecosystems. Ed. P.S. Cranston. Melbourne, CSIRO

Norton RA, Williams DD, Hogg ID, Palmen SC (1988) Biology of the orbatid mite: Mucronorthrus nasalis (Acari: oribatida: trhypochthoniidae) from small cold water springbok in eastern Canada. Canadian Journal of Zoology, 66, 622-629

Orth DJ, Maughan OE (1983) Microhabitat preferences of benthic fauna in a woodland stream. Hydrobiologia, 106, 157-168

Pudmenzky A, Marshall JC, Choy SC (1998) Preliminary application of artificial neural network model for predicting macroinvertebrates in rivers. Freshwater Biological Monitoring Report No. 9, The State of Queensland, Department of Natural Resources, Rocklea

Quinn JM, Hickey CW (1994) Hydraulic parameters and benthic invertebrate distributions in two gravel bed New Zealand rivers. Freshwater Biology 18, 521-528

Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagation errors. Nature 323, 533-536

Reynoldson TB, Norris RH, Resh VH, Day KE, Rosenberg DM (1997) The reference condition: a comparison of multimetric and multivariate approaches to assess water-quality impairment using benthic macroinvertebrates. Journal of the North American Benthological Society 16(4), 833-852

Schleiter IM, Borchardt D, Wagner R, Dapper T, Schmidt KD, Schmidt HH, Werner H (1999) Modelling water quality, bioindication and population dynamics in lotic ecosystems using neural networks. Ecological Modelling 120, 271-286

Simpson S, Norris R, Barmuta L, Blackman P (1997) Australian River Assessment System – National River Health Program Predictive Model Manual (first draft). CRC Freshwater Ecology, University of Canberra, Canberra, ACT

Smith BJ (1996) Identification keys to Families and Genera of Bivalve and Gastropod Moluscs foun in Australian Inland waters. Albury, NSW. CRC for Freshwater Ecology Identification Guide No.6

Statzner B, Gore JA, Resh VH (1988) Hydraulic stream ecology: observed patterns and potential applications. Journal of North American Benthological Society, 7, 307-360

Strahler AN (1957) Quantitative analysis of watershed geomorphology. Transactions of the American Geophysical Union, 38, 913-920

Suter PJ (1996) Baetidae. In: Mayfly Nymphs of Australia. A guide to Genera. Eds. J.C. Dean and P.J. Suter. Albury, NSW. CRC for Freshwater Ecology Identification Guide No.7

Walley WJ, Fontama VN (1998) Neural network predictors of average score per taxon and number of families at unpolluted river sites in Great Britain. Water Research 32(2), 613-622

Willoughby LG, Mappin RG (1988) Distribution of Ephemerella ignita (Ephemeroptera) in streams: The role of pH and food resources. Freshwater Biology, 19, 145-155

Wright JF (1995) Development and use of a system for predicting the macroinvertebrate fauna in flowing waters. Australian Journal of Ecology, 20, 181-197

Part III

Prediction and Elucidation of River Ecosystems

Chapter 12

Prediction and Elucidation of Population Dynamics of the Blue-green Algae *Microcystis aeruginosa* and the Diatom *Stephanodiscus hantzschii* in the Nakdong River-Reservoir System (South Korea) by a Recurrent Artificial Neural Network

K.-S. Jeong · F. Recknagel · G.-J. Joo

12.1 Introduction

Ecological modeling is an interdisciplinary branch in ecology. A model synthesized from adequate laboratory and field data can explain observed patterns and predict future ecosystem behaviors (Odum 1983; Krebs 1994). For accomplishing both objectives, it is necessary to use models that adequately address the uncertainty and complexity of ecosystems. Artificial Neural Networks (ANN) have been demonstrated to successfully model non-linear and complex phenomena. They have been used in aquatic ecology (e.g. Recknagel, 1997; Brosse et al., 1999), medicine, linguistics, and social sciences (Bullinaria 1997; Blom et al. 1999; Carson et al. 1999; Young et al. 2000).

ANN can function both as predictors and classifiers of temporal and spatial ecosystem patterns (e.g. Lek et al. 1996; Recknagel et al. 1997; Chon et al. 2000). Applications of scenario and sensitivity analyses have demonstrated explanatory capabilities of ANN (e.g. Recknagel and Wilson 2000; Jeong et al. 2001a). Highly complicated freshwater ecosystems can thus be elucidated to a certain extent by the ANN approach.

ANN may have good applicability in lotic and lentic freshwater ecosystems, which are distinguished primarily by the degree of water flow (see Burt 1992). Phytoplankton often is the major primary producer in these systems; it can exhibit very different population and community aspects in rivers compared to lakes (Reynolds 1992). Rivers with a high degree of flow regulation have even greater complexity and different community features (Stober and Nakatani 1992; Joo et al. 1997). Great efforts have been undertaken in modeling phytoplankton dynamics by heuristic and deterministic approaches (e.g. Kamp-Nielson 1978;

Reynolds 1984; Sommer et al., 1986; Kromcamp and Walsby 1990) and much has been achieved in understanding the ecology of this community. However, Recknagel (1997) and Jeong et al. (2001a) demonstrated that inductive modelling of phytoplankton dynamics by means of ANN can result in more realistic, holistic models by exploring the information content and complexity of aquatic time series. Jeong et al. (2001a) successfully applied recurrent neural networks (RNN) for modelling phytoplankton dynamics in the Nakdong River ecosystem of Korea.

Phytoplankton becomes a concern to the society when it forms a dense growth at the water surface, known as an algal bloom. Blooms frequently are observed in lowland rivers throughout the world. However, studies on mechanisms of algal bloom formation in rivers have received less attention than in lakes and reservoirs. Serious water quality problems in the Nakdong River are associated with blooms of *Microcystis aeruginosa* during hot summers, and of *Stephanodiscus hantzschii* from winter to early spring (Ha 1999). In this study, annual dynamics of these two algal species in the lower Nakdong River were modeled by RNN. Results of the study improved our understanding of factors controlling formation of algal blooms in regulated river systems.

12.2 Description of the Study Site

The Nakdong River basin is situated in the southeastern part of South Korea (35° to 37° N, 127° to 129° E) (Fig. 12.1). South Korea experiences four distinct seasons, and is characterized by heavy rainfall during the monsoon season and several typhoon events. The annual mean precipitation across the river basin is about 1,200 mm, but more than 50% of the annual rainfall is concentrated during summer (June–August). The annual mean water temperature at the study site was 13.7ºC. The mean water temperature was 2.2ºC during the coldest month (January), and 25.9ºC in August, the warmest month.

The main channel of the river is 526 km long, and the catchment area occupies about 25% of the whole country, covering an area of 23,817 km^2. The Mulgum station of the Nakdong River, from which data for the model were collected, is situated 27.4 km upstream of the estuarine dam at the river mouth, and has a maximum water depth of ~11 m, a mean depth of ~4 m, and a river width of 250-300 m.

Over 10 million people depend on the river for their drinking, agricultural, and industrial water supply. The Nakdong River has 4 multi-purpose dams and an estuarine dam. Physical alterations, industrialization, and urbanization have accelerated eutrophication of the lower part of the river (Kim et al. 1998).

12.3
Materials and Methods

12.3.1
Data Collection and Analysis

Environmental and limnological parameters (Table 12.1) were measured over a five-year period (1994-1998). Precipitation data were obtained from 5 representative meteorological stations (Andong, Daegu, Hapcheon, Jinju, and Miryang) within the Nakdong River basin.

River flow data were obtained from the Flood Control Center. Irradiance and evaporation data were collected from the Busan Local Meteorological Station, which is closest to the river study site.

Weekly water samples were collected at a depth of 0.5 m and the following water quality parameters were measured: water temperature, Secchi transparency, pH, turbidity, concentrations of dissolved oxygen (DO), nitrate (NO_3^--N), ammonia (NH_4^+-N), phosphate (PO_4^{3-}-P), dissolved silica (SiO_2), chlorophyll *a* (chl. *a*), phytoplankton biovolume, and zooplankton abundance. Water temperature and DO (mg L^{-1}) were determined with a YSI Model 58 meter; Secchi transparencies were determined with a 20-cm disk; pH was measured with an Orion Model 250A meter; and turbidity (NTU) was detected by Model 11052 Turbidimeter. Water samples were filtered using 0.45 μm Whatman GF/C glass filters to determine nutrient concentrations. The filtrates were frozen and analyzed by a QuikChem Automated Ion Analyzer (NO_3^--N, No. 10-107-04-1-O; NH_4^+-N, No. 10-107-06-1-B; PO_4^{3-}-P, No. 10-115-01-1-B; SiO_2, No. 10-114-27-1-A). Chlorophyll *a* concentrations were determined spectrophotometrically after extraction, using methods described by Wetzel and Likens (1991).

Phytoplankton samples were collected and immediately preserved with Lugol's solution. Species were identified by means of a Nikon light microscope (×1,000) and the following taxonomic references: Foged (1978), Cassie (1989), and Round et al. (1990). Phytoplankton was enumerated using an inverted microscope (ZEISS, ×400) by the sedimentation method after Utermöhl (1958). The biovolume of individual species was estimated from mean cell dimensions and the cellular shape of each species, according to Wetzel and Likens (1991). Mean cell biovolumes were based on individual cell volume calculations of 10 to 25 cells.

Zooplankton was collected from a depth of 0.5 m using a 3.2 L Van Dorn water sampler until a total of 8 L of water was obtained. Water samples were filtered through a 35-μm net, and the retained zooplankton was preserved with 10% formalin (final concentration: 4%). Macrozooplankton (almost exclusively *Copepoda* and *Cladocera*) was counted with an inverted microscope at ×25-50 magnification. Microzooplankton (mostly *Rotifera*) was counted with an inverted microscope at ×100-400 magnification. Zooplankton taxa were identified to genus

or species (except for juvenile *Copepoda*) using Koste (1978), Smirnov and Timms (1983), and Einsle (1993).

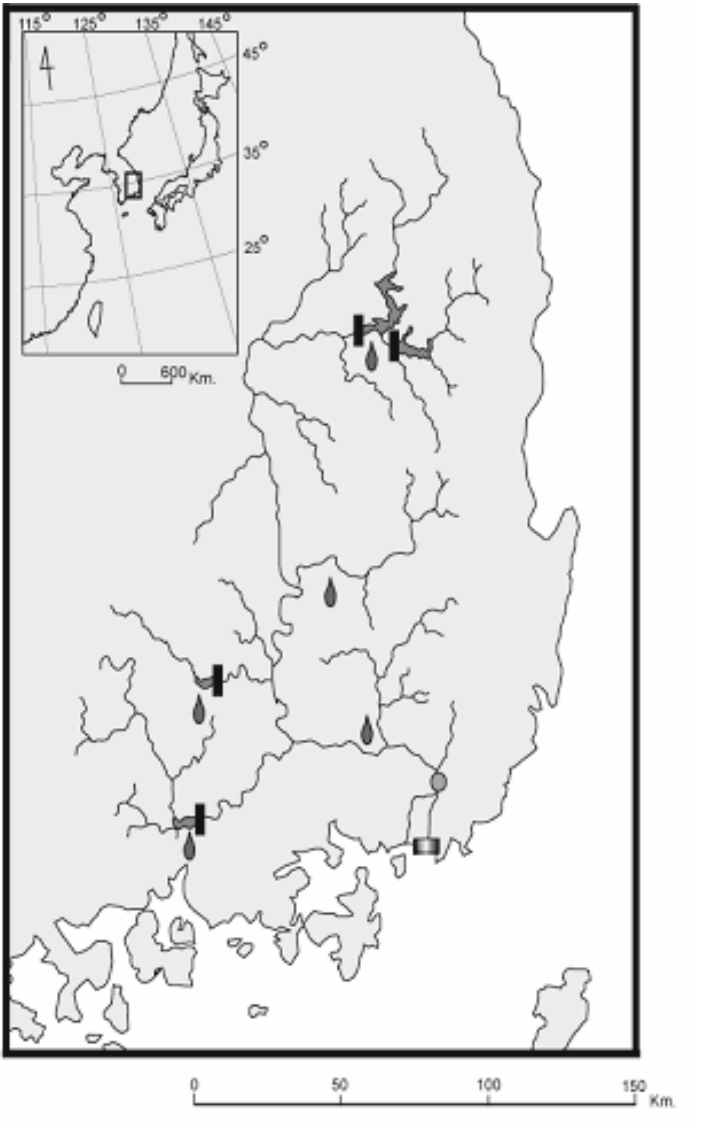


Fig. 12.1. Map of the Study Site. ▮ Multi-purpose dams; ▭ estuarine barrage ; ♦ rainfall gauging station; ● study site (Mulgum, RK 27).

12.3.2
Modelling the Phytoplankton Dynamics

The architecture of RNN (Fig. 12.2) was used to model the dynamic behavior of *M. aeruginosa* and *S. hantzschii* in the Nakdong River. Vectors of external inputs with time lags by up to 7 days were used to better explore seasonal trends in the time-series, as previously suggested for aquatic modelling by Chon et al. (2000), Jeong et al. (2001a), and Walter et al. (2001).

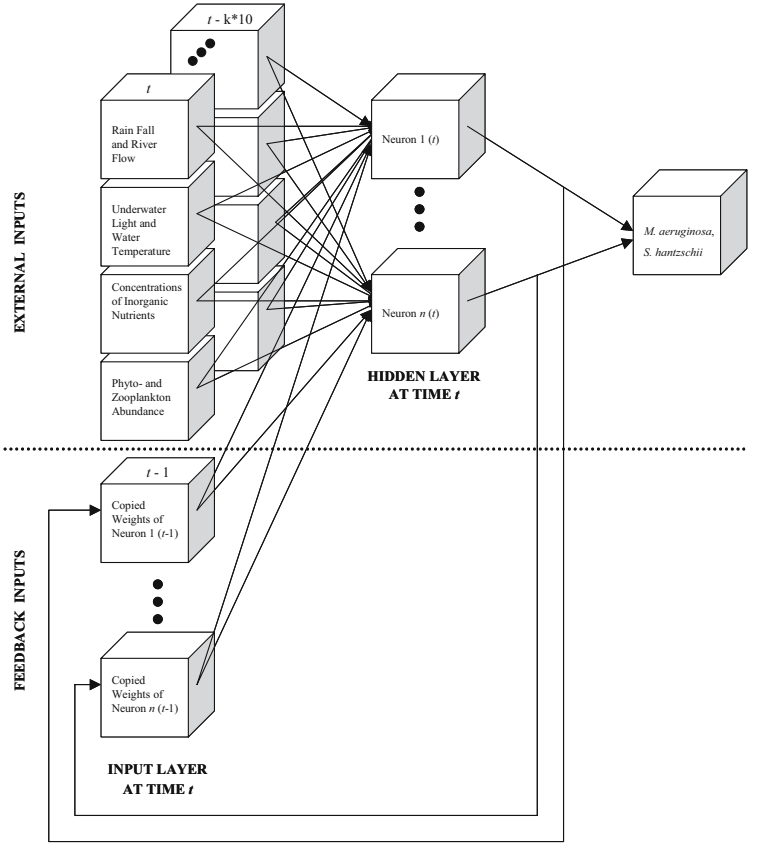


Fig. 12.2. Architecture of the Recurrent Neural Network Adopted in this Study.

RNN (Pineda 1987) are comparable to the deterministic modelling paradigm where the system state at time *t* is calculated by the means of system states at time (*t*-1) (see Recknagel 2001). Assuming that the weights of neurons of the hidden layer represent the "hidden" state of the system, copied weights of time (*t*-1) are considered as feedback inputs for the determination of weights of neurons at time

t. RNN are applicable to time-series modelling and forecasting (Connors et al. 1994).

While developing the RNN models, input and output data from the study site for 1995 to 1998 were used for training, and data from 1994 were used for validation (Table 12.1). Among the investigated parameters, chl. *a* was not used, in order to avoid autocorrelation between input and output variables. One hidden layer was selected for all the applications, and the numbers of nodes and neurons in the hidden layer were selected as control settings to find optimum training and prediction results by varying from 2 to 22. The hyperbolic tangent function was used to estimate the activation levels of both hidden and output layers, and the momentum was set at 0.7.

Tab. 12.1. Parameters used as input and output variables in the neural network optimum training and prediction results by varying from 2 to 22. The hyperbolic tangent function was used to estimate the activation levels of both hidden and output layers, and the momentum was set at 0.7.

Division	Categories	Variables	Unit
Input variables	Meteorological	Irradiance	MJ m^{-2} d^{-1}
	Hydrological	*Precipitation*	mm d^{-1}
		Discharge	CMS d^{-1}
		Evaporation	mm d^{-1}
	Physical	Water temperature	ºC
		Secchi depth	cm
		Turbidity	NTU
	Chemical	pH	
		DO	mg L^{-1}
		Nitrate-N	mg L^{-1}
		Ammonia-N	mg L^{-1}
		Phosphate-P	µg L^{-1}
		Dissolved silica	mg L^{-1}
	Biological	Rotifera	Ind. L^{-1}
		Cladocera	Ind. L^{-1}
		Copepoda	Ind. L^{-1}
Output variables	Biological	*M. aeruginosa*	µm^3 mL^{-1}
		S. hantzschii	µm^3 mL^{-1}

Training and validation were conducted by means of daily interpolated and averaged data. Fifteen trials on every time-vector were conducted with training iterations of 1,100, and the best-predicting models were selected based on the output variable based on a linear regression coefficient for every composed model. When a model was selected, additional training (1,000 iterations) was done to improve the network performance.

12.3.3
Neural Network Validation and Knowledge Discovery on Algal Succession

Model validation was based on visual comparisons between observed and simulated output values. With the best performing RNN, two types of sensitivity analyses as described in Jeong et al. (2001a) were implemented: 'Most Influencing Parameter (MIP)' and 'Sensitivity on Wide-ranged Disturbance (SWD)'.

The network was disturbed by ±1 to 2 SD for the sensitivity analyses. According to Zar (1984), ±1 SD represents commonly occurring variation, and ±2 SD covers about 95% of total data variation. The sensitivity analysis with ±1 SD can explain general conditions, while disturbance of ±2 SD may suggest specific and infrequent interactions between algal species and input variables. The results of sensitivity analyses were interpreted compared with known ecological information. All RNN models were developed by means of the neural network shell NeuroSolutions 3.0 (NeuroDimension, 1999).

12.4
Results and Discussion

12.4.1
Limnological Aspects and Plankton Dynamics in the Lower Nakdong River

Time series data from the lower Nakdong River indicate hypertrophic conditions and distinct annual and seasonal variability (Table 12.2). Due to the increased flow regulation, the ecosystem has been modified to become a river-reservoir hybrid (Joo et al. 1997). Construction of an estuarine barrage in conjugation with a water intake has increased water retention time and accelerated eutrophication in the lower 50 km of the river. Depending on the total amount of rainfall during the summer, the river exhibits distinctive seasonal characteristics. Rainfall patterns during the summer monsoon and typhoon events drive changes in physical-chemical parameters in the lower Nakdong River (Park 1998; Lee et al. 1999).

Rotifera dominated the zooplankton community of the river from 1994 to 1998, while *Cladocera* and *Copopoda* were much less abundant. Zooplankton populations did not display significant inter-annual variation. However, there were "clear water phases" in early spring and autumn, as earlier documented and attributed to macrozooplankton grazing of phytoplankton (Kim et al. 1998; Kim et al. 2001).

Table 12.2. Limnological characteristics of the lower Nakdong River for five years (1994-1998). *, means ± SD (n = 263; 52-53 in each year).

Division	Parameters	Unit	Mean±SD					
			5 years'	1994	1995	1996	1997	1998
Meteorological	Irradiance	MJ m^{-2} day^{-1}	12.8±6.5*	14±7	14±6	12±6	13±6	12±6
	Air temperature	ºC day^{-1}	15±8	16±9	15±8	15±8	15±8	16±8
Hydrological	Precipitation	mm day^{-1}	974±306	765	841	1007	1352	1670
	Discharge	CMS	567±714	399±79	466±358	488±480	686±825	794±1184
	Evaporation	mm day^{-1}	3±2	4±2	3±2	3±2	3±2	3±1
Physical	Water temperature	ºC	17±9	20±10	16±10	17±10	18±9	17±8
	Secchi depth	cm	74±25	72±22	75±20	74±22	74±32	74±23
	Turbidity	NTU	18±54	20±64	12±35	9±9	19±38	27±91
Chemical	pH		8.4±0.8	8.7±0.9	8.3±0.6	8.4±0.7	8.5±0.8	8.0±0.8
	DO	mg L^{-1}	10.8±4.0	9.9±3.8	11.4±3.6	11.9±3.9	10.2±4.5	10.5±3.4
	Conductivity	μs cm^{-1}	349±128	312±92	405±118	396±114	374±146	250±76
	Alkalinity	mg $CaCO_3$ L^{-1}	57±17	55±13	66±13	67±13	58±17	41±9
	Nitrate-N	mg L^{-1}	2.7±1.0	1.8±0.9	2.5±1.0	2.3±1.0	3.3±0.8	3.2±0.5
	Ammonia-N	mg L^{-1}	0.6±0.7	0.3±0.3	0.8±0.8	0.7±0.6	0.3±0.3	0.8±1.0
	Phosphate-P	μg L^{-1}	34.7±25.2	33.1±22.1	34.3±25.2	20.5±15.2	32.7±23.0	52.8±27.9
	Silica	mg L^{-1}	4.3±3.8	3.6±2.3	2.6±2.8	3.0±2.3	4.6±4.2	7.5±4.4
Biological	Rotifera	ind. L^{-1}	1644±3250	1241±2086	1285±1764	1021±1274	3046±5713	1304±1747
	Cladocera	ind. L^{-1}	91±311	25±58	201±588	71±176	79±140	30±61
	Copepoda	ind. L^{-1}	60±151	23±43	65±147	43±67	109±251	36±62
	M. aeruginosa.	X10^6 μm^3 mL^{-1}	2.84±12.34	5.34±18.81	1.42±3.08	3.66±11.51	3.64±15.82	0.15±0.39
	S. hantzschii	X10^6 μm^3 mL^{-1}	15.10±24.14	12.97±26.74	17.24±29.42	20.89±27.11	10.22±22.88	9.50±11.48
	Chlorophyll *a*	μg L^{-1}	50.2±91.5	84.7±178.5	65.5±74.7	48.5±49.2	37.5±80.6	28.0±26.4

Overall phytoplankton dynamics were strongly influenced by magnitudes and timing of *M. aeruginosa* and *S. hantzschii* blooms. Annual average biovolumes of *M. aeruginosa* and *S. hantzschii* peaked in years with low annual precipitation. *M. aeruginosa* especially proliferated during the extreme drought of 1994, while the peak biovolume of *S. hantzschii* occurred in the winter of 1996. Both *M. aeruginosa* and *S. hantzschii* accounted for 80% of the phytoplankton abundance as a result of fast growth in the summer and winter, respectively. During blooms, these two species accounted for more than 90% of the algal abundance.

Microcystis spp. rarely forms blooms in flowing water systems except pool-like and sluggish rivers (see Reynolds 1992). Even though centric diatoms such as *S. hantzschii* were found widely in river systems (Lack 1971; Moss and Balls 1989; Köhler 1994; Murakami 1998), there have been almost no reports of winter *Stephanodiscus* blooms. Ha et al. (1999) reported that hydrologic stagnation in the Nakdong River influenced phytoplankton dynamics. In particular, the *Microcystis* bloom formation was directly related to the importance of hydrodynamics and nutrient loading. The *Stephanodiscus* proliferation may be due to combined factors such as cold temperature, low flow, and high availability of dissolved silica.

12.4.2
Configuring the Neural Network Architecture and Predictability

Network training (Table 12.3) was done with various time-delayed input vectors. The number of nodes in all time vectors was between 9 and 21 when the boundary was given between 2 and 22, except for the case of a 1-day-delay (case 10), which satisfied criteria suggested from Hecht-Nielsen (1987). Using the 4-year data set, there was little tendency for decreases of node number and mean squared error (MSE) when the time-delay was increased. The MSE for network training decreased as TDL increased by month with a fixed number of hidden layer nodes (see Chon et al., 2000). However, among 8 time vectors, 4-day-delay inputs gave a significant negative correlation between node number and MSE. Jeong et al. (2001a) predicted time-series algal biomass in the lower Nakdong River using a model with 3-day-delayed inputs; this was related to water residence time. According to the ecological input data (i.e. number of exemplars, input parameters, and their unseen relationships), an adequate TDL could be selected variably.

Table 12.3. Node numbers and MSE for each time-delayed vector with 1,100 iterations , correlation coefficients between node number and MSE for each time vector , and best-predicting network for 1994 algal dynamics.

Cases	No-delay		1-day-delay		2-day-delay		3-day-delay		4-day-delay		5-day-delay		6-day-delay		7-day delay	
	Nu.	MSE	Nu.	MSE	Nu.	MSE	Nu.	MSE	Nu.	MSE	Nu.	MSE	Nu.	MSE	Nu.	MSE
1	19	0.003	14	0.003	16	0.002	16	0.003	11	0.004	16	0.003	12	0.004	18	0.003
2	15	0.004	15	0.004	13	0.003	18	0.002	19	0.003	19	0.006	13	0.004	15	0.002
3	17	0.004	20	0.002	14	0.003	15	0.004	18	0.003	21	0.003	17	0.003	17	0.002
4	20	0.002	18	0.002	14	0.002	21	0.006	16	0.004	21	0.004	18	0.003	18	0.002
5	19	0.003	17	0.006	21	0.002	15	0.003	21	0.002	21	0.003	15	0.002	11	0.003
6	13	0.003	13	0.004	19	0.002	21	0.004	16	0.004	21	0.006	21	0.003	20	0.005
7	20	0.005	15	0.004	15	0.002	16	0.002	18	0.003	19	0.002	20	0.006	21	0.002
8	20	0.002	18	0.006	20	0.001	19	0.003	20	0.003	15	0.002	20	0.003	19	0.004
9	12	0.007	20	0.003	16	0.003	21	0.002	21**	0.003	21	0.003	18	0.003	16	0.003
10	15	0.004	22	0.002	18	0.002	14	0.005	18	0.003	21	0.002	10	0.004	13	0.003
11	20	0.003	22	0.007	19	0.002	15	0.004	21	0.003	11	0.005	15	0.004	18	0.003
12	15	0.004	15	0.004	15	0.002	21	0.002	14	0.003	12	0.004	21	0.006	15	0.003
13	20	0.003	18	0.003	20	0.002	16	0.003	19	0.003	17	0.002	20	0.004	6	0.005
14	11	0.001	20	0.003	16	0.002	15	0.004	11	0.006	21	0.006	17	0.002	20	0.005
15	16	0.004	20	0.007	21	0.003	15	0.004	21	0.003	9	0.006	21	0.002	20	0.002
Max.	20	0.007	22	0.007	21	0.003	21	0.006	21	0.006	21	0.006	21	0.006	18	0.003
Min.	11	0.002	13	0.002	13	0.001	14	0.002	11	0.002	9	0.002	10	0.002	11	0.002
Mean		0.004		0.004		0.002		0.003		0.003		0.004		0.004		0.003
r*	-0.58 ($0.02 > p > 0.01$)		0.01 ($p > 0.50$)		-0.29 ($0.50 > p > 0.20$)		-0.15 ($p > 0.50$)		-0.78 ($0.001 > p$)		-0.24 ($0.50 > p > 0.20$)		0.15 ($p > 0.50$)		-0.39 ($0.20 > p > 0.10$)	

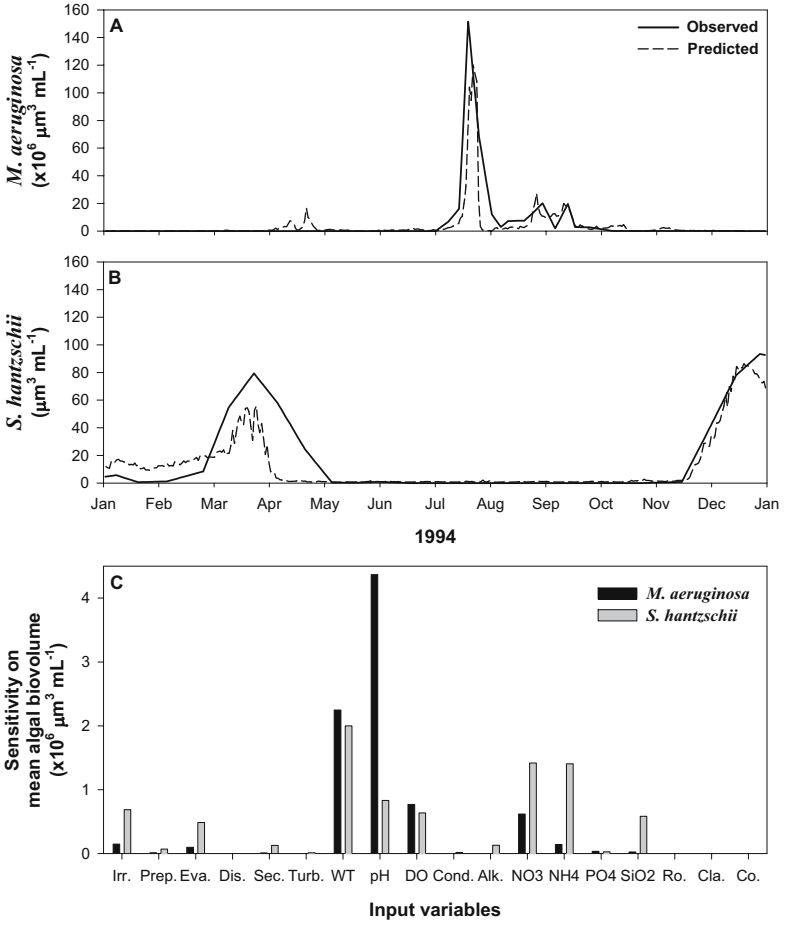


Fig. 12.3. Prediction results for *M. aeruginosa* (A) and *S. hantzschii* (B) and the MIP analysis (C)

The best-predicting network was obtained from the 4-day-delayed input vector. It displayed the best predictability when initially trained at 1,100 iterations. After additional 1,000 iterations, the final MSE reached 0.0017 and the results of prediction were quite good (Fig. 12.3). The model effectively predicted species dynamics in 1994 (Fig. 12.3A). The timing of peak biovolume of both species was well recognized, even though there was some under-estimation in the late bloom of *S. hantzschii*. In the case of *M. aeruginosa*, the model produced a small spring peak in April, but the actual magnitude was much smaller. Recknagel (1997),

Chon et al. (2000), and Jeong et al. (2001a) previously documented the predictability of ANN for species abundance and succession of freshwater algae and macro-invertebrates. In this study, the Time-Delayed Recurrent Neural Network (TDRNN) recognized distinct seasonal abundance and succession of phytoplankton species as typical for the lower Nakdong River.

12.4.3 Elucidation of Ecological Hypotheses

Based on the Most Influencing Parameter sensitivity analysis (MIP) of the validated RNN, both phytoplankton species were influenced largely by water temperature and pH (Fig. 12.3c). Dissolved Inorganic Nitrogen (DIN) was important for the biovolume changes of *S. hantzschii*, but had relatively minor effects on *M. aeruginosa*. Meteorological events, hydrological regimes, and zooplankton abundances had less impact on dynamics of the phytoplankton. The patterns observed in the phytoplankton are consistent with the conclusion of Joo et al. (1997), that the lower Nakdong is a reservoir-like ecosystem. Phytoplankton also is influenced during a short period in the summer rainy season, when there is a sudden increase of discharge (Ha 1999), and a base flow that continues from fall to late spring (Park 1998).

The occurrence of both bloom-forming species was previously found to be correlated with increased pH, water temperature, and nutrient availability in lakes and reservoirs (Reynolds 1984; Harris 1986; Sommer et al. 1986; Shapiro 1990). Results of the RNN-based sensitivity analysis correspond with these previous findings and indicate that short-term dynamics of bloom-formation in river-reservoir systems like the Nakdong River are mainly driven by physical-chemical parameters such as temperature and pH, while long-term trends are basically determined by the hydrological regime.

Sensitivity on Wide-ranged Disturbance (SWD) for the input variables revealed information on relationships between input variables and both species (Fig. 12.4). The response differed among the two species considered. Irradiance, evaporation, Secchi depth, DO, conductivity, nitrate, phosphate, and dissolved silica concentration influenced the increase or decrease of phytoplankton, while rainfall, discharge, turbidity, ammonia concentration, and zooplankton abundances did not have any recognizable effects. As we changed the range of disturbance for evaporation and water temperature, both species exhibited dynamic variations.

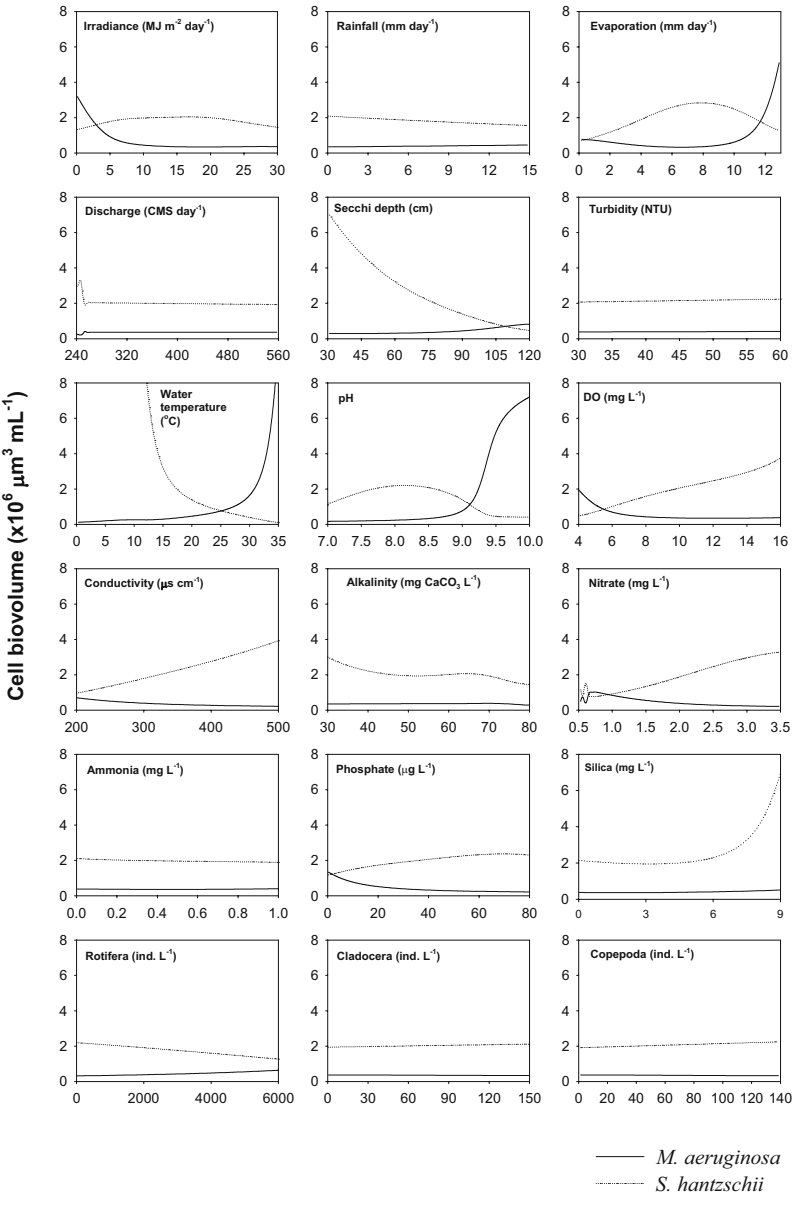


Fig. 12.4. Results of SWD analysis for the trained model (sensitivity analysis).

12.4.3.1 Microcystis aeruginosa

Shapiro (1990) presented several hypotheses to explain dominance of blue-greens in the phytoplankton of north temperate lakes. These included positive responses to high water temperature, low light, low N/P ratios, buoyancy, zooplankton grazing, and the CO_2 - pH complex. The results of SWD analysis (Fig. 12.4) revealed relationships between *M. aeruginosa*, water temperature and pH consistent with this hypothesis. As stated by Shapiro (1990), blue-green algae are most competitive at temperatures exceeding 20°C. Results of the SWD indicated that *M. aeruginosa* starts to grow at 20 °C and then explosively augments the phytoplankton at temperatures in excess of 30 °C in the lower Nakdong River.

Changes in pH and CO_2 concentration are important factors controlling cyanobacteria dominance (Harris, 1986). At pH values greater than 8.5, exponential growth occurs. King (1970) and Shapiro (1984) suggested that physiological adaptations of blue greens enable them to out-compete eukaryotic algae at high pH and/or low CO_2. Reynolds (1986) documented that this phenomenon held true for both *M. aeruginosa* and *Anabaena flos-aquae*. In the case of the lower Nakdong River, Ha et al. (1999) reported that seasonal changes of pH are important for the middle phase of proliferation of blue-greens, rather than stimulating their initial growth. Results of the SWD on the validated RNN model indicate a strong correlation between increasing dominance of *M. aeruginosa* in the lower Nakdong River and increased pH, consistent with field observations (Talling 1976) and controlled experiments (Shapiro 1984) on other freshwater systems.

Another factor found by SWD to be strongly correlated with the proliferation of *M. aeruginosa* was evaporation, which is a function of the river's heat budget and therefore its water temperature (Yoon 1998). Severe evaporation during a drought can be linked to a decline of water flow, which causes high water stagnation in the lower Nakdong River. Jeong et al. (2001a) documented a similar influence of evaporation on the changes of chl *a* concentrations through time-series ANN model application.

Complex interactions among environmental parameters appear to be responsible for the severe bloom events in the lower Nakdong River. With elevated water temperatures, low discharge rates, and high irradiance, blooms of *Microcystis* spp. are stimulated in this river. In addition, high nutrient concentrations are a necessary pre-requisite for out-breaks of blue green algal blooms (Ha et al. 1999). After the initial phase of algal proliferation, pH is important for selecting the dominant species.

12.4.3.2 Stephanodiscus hantzschii

Stephanodiscus hantzschii blooms in the lower Nakdong River could be explained by hydrodynamic factors. The results of SWD indicated that water temperature,

Secchi depth, pH, and dissolved silica were strongly related to the dynamics of the species (Fig. 12.4). *Stephanodiscus* is known to prefer low water temperatures, and several European eutrophic rivers had blooms of this genus in the winter (temperatures around 15 °C) (Descy et al. 1987). Ha (1999) reported that *S. hantzschii* blooms in the lower Nakdong River occurred at much lower temperatures (4-8 °C). In this SWD study, when the water temperature exceeded 7-8 °C, biovolume of *S. hantzschii* sharply decreased.

Because diatoms use dissolved silica to generate frustules, the concentration of SiO_2 is an important determinant of their growth (Round et al. 1990). An enclosure experiment conducted by Ha et al. (1998) and Ha and Joo (2000) indicated that *S. hantzschii* was dominant at low SiO_2 (1–1.5 mg L^{-1}), while *Fragilaria crotonensis* and *Synedra acus* had a competitive advantage in SiO_2-added enclosures. From the field data, ANN could recognize the importance of dissolved silica, which was similar to that indicated by the experimental results.

Stephanodiscus hantzschii abundance increased at pH levels around 7.5-8.5 and started to decrease after pH 9.0. Similar to the case of *M. aeruginosa*, this situation can be interpreted by the CO_2-pH complex. Most algal species, except blue-greens, are sensitive to dissolved carbon dioxide in the water, because they are unable to utilize the other dissolved forms that occur at higher pH. The decrease of *S. hantzschii* at higher pH values could be explained by this phenomenon.

12.5 Implications on Ecological Informatics for Limnology

Natural ecosystems are distinctly non-linear, dynamic and complex. Powerful mathematical and computational techniques are required to elucidate and predict driving forces and processes underlying extreme ecosystem behaviors such as algal bloom events (Straskraba 1994). As shown in this study, artificial neural networks prove to be one suitable computational technique for these purposes. However, the newly emerging discipline of ecological informatics provides a variety of computational techniques such as fuzzy logic, cellular automata, evolutionary algorithms and adaptive agents (e.g. Fielding 1999; Whigham and Recknagel 2001a; Whigham and Recknagel 2001b; Bobbin and Recknagel 2001; Jeong et al. 2001b; Recknagel 2001). Paired with growing power of computers these techniques extent, complement, reinforce or hybridise ecological modeling techniques towards more realistic modelling of limnological phenomena at different levels of organization and complexity (see Fig. 12.5).

Hybrid architectures of empirical models as suggested by Medsker (1996) may further encourage inter-disciplinary research between ecology and computer science. For example, although the results of this study can stand alone by their good performance, they also can serve as information for developing Cascade Artificial Neural Networks (CANN), linking together the various study sites (e.g. between upper and lower river segments). Neuro-Genetic Learning (NGL), which

evolves either neural network's architecture or its weights, is another example for using informatics in the prediction of phytoplankton dynamics in a time series (Jeong et al. 2001b). With its sophisticated methods, ecological informatics is highly suitable for searching out and predicting ecosystem dynamics.

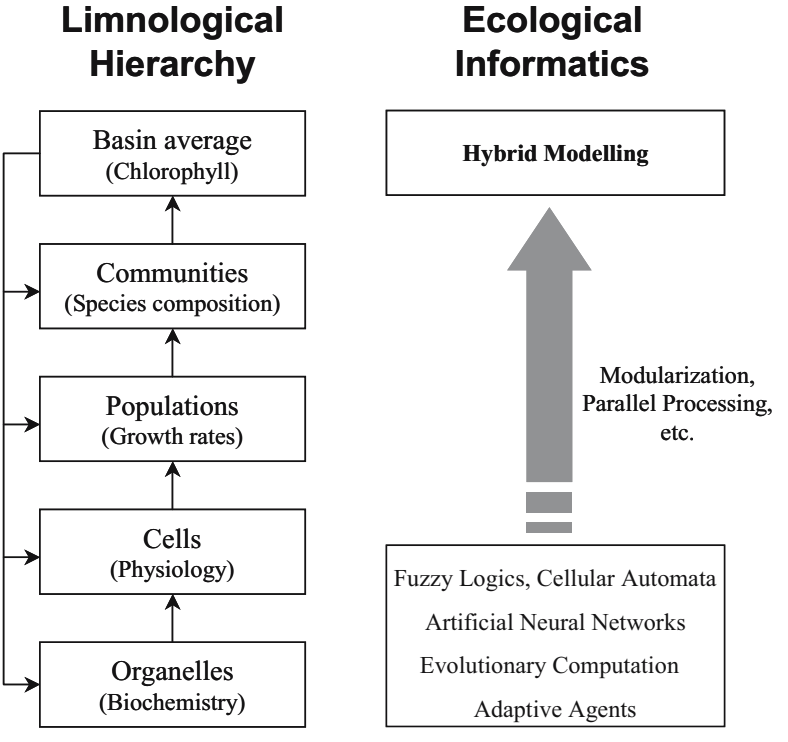


Fig. 12.5. Contribution of Ecological Informatics to Hybrid Modelling of Limnological Phenomena at Different Levels of Organisation

12.6 Conclusions

Artificial neural networks were applied to the prediction and elucidation of two bloom forming algal species in the Nakdong river-reservoir system. The lower Nakdong River, which has characteristics of both rivers and reservoirs, represents a complicated system for algal bloom modeling. Yet, RNN proved capable not only to predict the distinct seasonal abundance and succession of *Microcystis*

aeruginosa and *Stephanodiscus hantzschii* but elucidate key driving variables by means of sensitivity analyses. Findings of the sensitivity analysis corresponded very well with existing theories on the ecology of these two algae species.

This study yields promising results for the application of machine learning to complex ecosystems such as regulated rivers. It encourages inter-disciplinary research between ecologists, modelers and computer scientists in the newly emerging area of ecological informatics in order to better understand and predict ecological phenomena at different levels of organization.

Acknowledgements

The authors thank Dr. H. W. Kim of Sunchon National University and Dr. K. Ha of Pusan National University (PNU) for their efforts in the analyses of plankton data. We appreciate Mr. S. B. Park and J. G. Kim for their helps with field sampling. This study was supported by the Institute of Environmental Technology and Industry (IETI) (Project No., 01-10-99-01-A-1). This is a contribution No. 22 of Nakdong River Research Programme in Limnology Lab., PNU.

References

Blom N, Gammeltoft S, Brunak S (1999) Sequence and structure-based prediction of eukaryotic protein phosphorylation sites. N. Mol. Biol., 294: 1351-1362

Bobbin J, Recknagel F (2001) Knowledge discovery for prediction and explanation of blue-green algal dynamics in lakes by evolutionary algorithms. Ecol. Modelling 146, 1-3, 253-262

Brosse S, Guégan JF, Tourenq JN, Lek S (1999) The use of artificial neural networks to assess fish abundance and spatial occupancy in the littoral zone of a mesotrophic lake. Ecol. Modelling, 120: 299-311

Bullinaria JA (1997) Modeling reading, spelling, and past tense learning with artificial neural networks. Brain Lang., 59: 236-266

Burt TP (1992) The Hydrology of Headwater Catchments. In: (Eds) P. Calow and G. E. Petts. The River Handbook: Hydrological and Ecological Principles. Vol. 1. Blackwell Scientific Publication, Oxford, 526 pp

Carson AD, Bizot EB, Hendershot PE, Barton MG, Garvin MK, Kraemer B (1999) Modeling career counselor decisions with artificial neural networks: predictions of fit across a comprehensive occupational map. J. Vocational Behav., 54: 196-213

Cassie V (1989) A Contribution to the Study of New Zealand Diatoms. J. Cramer, Berlin, 266 pp

Chon TS, Park YS, Cha EY (2000) Patterning of Community Changes in Benthic Macroinvertebrates Collected from Urbanized Streams for the Short Term Prediction by Temporal Artificial Neuronal Networks. 99-114. In: S. Lek and J. F. Guégan (Eds). Artificial Neuronal Networks: Application to Ecology and Evolution. Springer-Verlag, Berlin, 97-113

Connors J, Martin D, Atlas L (1994) Recurrent neural networks and robust time series prediction. IEEE T. Neural Networ., 5: 240-254

Descy JP (1987) Phytoplankton composition and dynamics in the River Meuse (Belgium). Arch. Hydrobiol., 78: 225-245

Einsle U (1993) Crustacea, Copepoda, Calanoidia and Cyclopoida. Susswasserfauna von Mitteleuropa, Vol. 8, Part 4-1, J. Fisher, Stuttgart, 208 pp

Fielding A (1999) An introduction to machine learning methods. In: (Ed) A. Fielding. Machine Learning Methods for Ecological Applications. Kluwer Academic Publishers, Massachusetts, 261 pp

Foged E (1978) Diatoms in Eastern Australia, J. Cramer, Berlin, 243 pp

Ha K, Joo GJ (2000) Role of silica in phytoplankton succession: an enclosure experiment in the downstream Nakdong River (Mulgum). Korean J. Ecol., 23: 299-307

Ha K, Cho EA, Kim HW, Joo GJ (1999) Microcystis bloom formation in the lower Nakdong River, South Korea: importance of hydrodynamics and nutrient loading. Mar. Freshwater Res., 50: 89-94

Ha K, Kim HW, Joo GJ (1998) The phytoplankton succession in the lower part of hypertrophic Nakdong River (Mulgum), South Korea. Hydrobiologia, 369/370: 217-227

Ha K (1999) Phytoplankton Community Dynamics and Microcystis Bloom Development in a Hypertrophic River (Nakdong River, Korea). Ph. D. dissertation. Pusan National Univ., Pusan, 140 pp

Harris GP (1986) Phytoplankton Ecology: Structure, Function and Fluctuation. Chapman and Hall, NY, 384 pp

Hecht-Nielsen R (1987) Neurocomputing. Addison-Wesley Publishing Co., NY, 433 pp

Jeong KS, Joo GJ, Kim HW, Ha K, Recknagel F (2001a) Prediction and elucidation of algal dynamics in the Nakdong River (Korea) by means of a recurrent artificial neural network. Ecol. Modelling, 146: 115-129

Jeong KS, Jang MH, Park SB, Cho GI, Joo GJ (2001b) Neuro-Genetic Learning to the algal dynamics: a preliminary experiment for the new technique to the ecological modelling. Proceeding of the Korean Environmental Science Society, pp 234-235

Joo GJ, Kim HW, Ha K, Kim JK (1997) Long-term trend of the eutrophication of the lower Nakdong River. Kor. J. Limnol., 30-supplement: 472-480

Kamp-Nielsen L (1978) Modelling the vertical gradients in sedimentary phosphorus fractions. Verh. Internat. Verein. Limnol., 20: 720-727

Kim HW, Joo GJ, Walz N (2001) Zooplankton dynamics in the hyper-eutrophic Nakdong River system (Korea) regulated by an estuary dam and side channels. Internat. Rev. Hydrobiol., 86: 127-143

Kim HW, Ha K, Joo GJ (1998) Eutrophication of the lower Nakdong River after the construction of an estuarine dam in 1987. Internat. Rev. Hydrobiol., 83: 65-72

King DL (1970) The role of carbon in eutrophication. J. Water Poll. Contr. Fed., 42: 2035-2051

Köhler J (1994) Origin and succession of phytoplankton in a river-lake system (Spree, Germany). Hydrobiologia, 289: 73-83

Koste W (1978) Rotatoria. Die Radertiere Mitteleuropes. Ein Bestimmungswerk begrunder von Max Voigt. 2nd ed. Borntrager, Stuttgart, Vol. 1, Textband 673 pp., Vol. 2. Tafelband 234 pp

Krebs CJ (1994) Ecology: the Experimental Analysis of Distribution and Abundance. Harper Collins College Publishers, NY, 801 pp

Kromkamp J, Walsby AE (1990) A computer model of buoyancy and vertical migration in cyanobacteria. J. Plankton Res., 12: 161-183

Lee SK, Choi SH, Kim HW, Ha K, Joo GJ (1999) Inter-annual variability of nutrient loadings in the lower Nakdong River, Mulgum, Korea. Acta Hydrobiol. Sinica, 23: 17-23

Lek S, Delacoste M, Baran P, Dimopoulos I, Lauga J, Aulagnier S (1996) Application of neural networks to modelling nonlinear relationships in ecology. Ecol. Modelling, 90: 39-52

Medsker LR (1996) Microcomputer applications of hybrid intelligent systems. J. Networ. Comput. Appl., 19: 213-234

Moss B (1998) Ecology of Fresh Waters: Man and Medium, Past to Future. 3rd ed. Blackwell Science, Oxford, 557 pp

Murakami T (1998) Flora and biomass of planktonic communities upstream of a river mouth dam in Japan. Internat. Rev. Hydrobiol., 83: 463-466

NeuroDimension (1999) NeuroSolutions: The Neural Network Simulation Environment, (Vers. 3.02 consultants level) and NeuroSolutions for Excel (Vers. 1.02)

Odum EP (1983) Basic Ecology. Saunders College Publishing, Florida, 613 pp

Paerl HW (1988) Nuisance phytoplankton blooms in coastal, estuarine, and inland waters. Limnol. Oceanogr. 33: 823-847

Park SB (1998) Basic Water Quality of the Mid to Lower Part of Nakdong River and the Influences of the Early Rainfall during Monsoon on the Water Quality. M. S. thesis. Pusan National Univ., Pusan, 104 pp. (in Korean)

Pineda F (1987) Generalization of backpropagation to recurrent neural networks. Phys. Rev. Lett., 19, 59, 2229-2232

Recknagel F (2001) Applications of machine learning to ecological modeling. Ecol. Modelling 146, 1-3, 303-310

Recknagel F, Wilson H (2000) Elucidation and prediction of aquatic ecosystems by artificial neuronal networks. In: (Eds) S. Lek and J. F. Guégan. Artificial Neuronal Networks: Application to Ecology and Evolution. Springer-Verlag, Berlin, 143-155

Recknagel F (1997) ANNA-Artificial Neural Network model for predicting species abundance and succession of blue-green algae. Hydrobiologia, 349: 47-57

Recknagel F, French M, Harkonen P, Yabunaka KI (1997) Artificial neural network approach for modelling and prediction of algal blooms. Ecol. Modelling, 96: 11-28

Reynolds CS (1984) The Ecology of Freshwater Phytoplankton. Cambridge University Press, NY, 384 pp

Reynolds CS (1986) Experimental manipulation of phytoplankton prediodicity in large limnetic enclosures in Blelham Tarn, English Lake District. In: (Eds) M. Munawar and J. F. Talling. Seasonality of Freshwater Phytoplankton. Junk, Dordrecht

Reynolds CS (1992) Algae. In: (Eds) P. Calow and G. E. Petts. The River Handbook: Hydrological and Ecological Principles. Vol. 1. Blackwell Scientific Publication, Oxford, 526 pp

Round FE, Crawford RM, Mann DG (1990) The Diatoms, Cambridge University Press, New York, 747 pp

Shapiro J (1984) Blue-green dominance in lakes: the role and management significance of pH and CO_2. Internat. Revue Ges. Hydrobiol., 69: 765-780

Shapiro J (1990) Current beliefs regarding dominance by blue-greens: the case for the importance of CO_2 and pH. Verh. Int. Verein. Limnol., 24: 38-54

Smirnov NN, Timms BV (1983) A revision of the Australian Cladocera (Crustacea). Records of the Australian Museum Supplement, 1: 1-132

Sommer U, Gliwicz ZM, Lampert W, Duncan A (1986) The PEG-model of seasonal succession of planktonic events in fresh waters. Arch. Hydrobiol., 106: 433-471

Stober QJ, Nakatani RE (1992) Water quality and biota of the Columbia River system. In: (Eds) C. D. Becker and D. A. Neitzel. Water Quality in North American River Systems, Battelle Press, Ohio, 51-83 pp

Straskraba M (1994) Ecotechnological models for reservoir water quality management. Ecol. Modelling, 74: 1-38

Talling JF (1976) The depletion of carbon dioxide from lake water by phytoplankton. J. Ecol., 64: 79-121

Utermöhl H (1958) Zur Vervollkommnung der Quantitativen Phytoplankton. Methodik. Mitt. Internat. Verein. Limnol., 9: 1-38

Walter M, Recknagel F, Carpenter C, Bormans M (2001) Predicting eutrophication effects in the Burrinjuck Reservoir (Australia) by means of the deterministic model SALMO and the recurrent neural network model ANNA. Ecol. Modelling 146, 1-3, 97-113

Wetzel RG, Likens GE (1991) Limnological Analyses. 2nd ed. Springer-Verlag, New York, 391 pp

Whigham PA, Recknagel F (2001) An inductive approach to ecological time series modeling by evolutionary computation. Ecol. Modelling 146, 1-3, 275-287

Whigham PA, Recknagel F (2001) Predicting chlorophyll-a in freshwater lakes by hybridising process-based models and genetic algorithms. Ecol. Modelling 146, 1-3, 243-251

Yoon YN (1998) Industrial Hydrology. Cheongmoongak Publishers, Seoul, 656 pp. (in Korean)

Young MT, Blanchard SM, White MW, Johnson EE, Smith WM, Ideker RE (2000) Using an artificial neural network to detect activation during ventricular fibrillation. Comput. Biomed. Res., 33: 43-58

Zar JH (1984) Biostatistical Analysis. 2nd ed. Prentice-Hall, NJ, 718 pp

Chapter 13

An Evaluation of Methods for the Selection of Inputs for an Artificial Neural Network Based River Model

G.J. Bowden · G.C. Dandy · H.R. Maier

13.1 Introduction

Artificial Neural Network (ANN) models are highly flexible function approximators, which have shown their utility in a broad range of ecological modelling applications. The rapid emergence of ANN applications in the field of ecological modelling can be attributed to their advantages over standard statistical approaches. Such flexibility provides a powerful tool for forecasting and prediction, however, the large number of parameters that must be selected only serves to complicate the design process. In most practical circumstances, the design of an ANN is heavily based on heuristic trial-and-error processes with only broad rules of thumb to guide along the way.

The main steps in the development of an ANN model include choice of performance criteria, division of data, data pre-processing, determination of model inputs, determination of network architecture, optimisation (training) and model validation (Maier and Dandy 2000a). One of the most important steps in this developmental process is the determination of the significant input variables. Where the potential number of input variables to an ANN is large and little *a priori* knowledge is available to suggest which subset of variables to include, the selection process is inherently difficult. In this paper, the step involving the determination of model inputs is considered in detail and a number of different methods are evaluated. As far as possible, all other steps in the ANN modelling process are held constant so that the various input determination techniques can be compared.

In the majority of ANN applications, practitioners give little attention to the task of input selection (Maier and Dandy 2000b). This is largely because ANNs belong to the class of data driven approaches, whereas conventional statistical methods are model driven (Chakraborty et al. 1992). In the latter, the model's

structure is determined first by using empirical or analytical approaches, before estimating the unknown model parameters. Data driven approaches are usually assumed to be able to determine which model inputs are critical. However, as pointed out by Maier and Dandy (2000b), presenting a large number of inputs to the ANN and relying on the network to determine the significant inputs, usually increases the network size. This results in certain disadvantages, such as an increase in the amount of data required to estimate the connection weights properly and a reduction in processing speed (Lachtermacher and Fuller 1994).

The input selection problem can be formulated as having a set of input variables to an ANN, and an output value which can be used to evaluate the fitness or merit of the network using those variables. For example, for the problem of forecasting cyanobacteria, the inputs to the ANN are causal variables such as turbulence (flow), water temperature, turbidity, colour and nitrogen and phosphorus concentrations. From these variables, a subset of inputs must be selected that yield the network of highest fitness. This subset of inputs forms an n-dimensional input vector $\boldsymbol{X}^n$, which can be used to forecast the concentration of cyanobacteria $\boldsymbol{Y}$. The aim of the ANN is to produce a generalised relationship of the form

$$\boldsymbol{Y}=f(\boldsymbol{X}^n) \tag{13.1}$$

It is a generalised relationship because the functional form of $f(\bullet)$ is not revealed explicitly but rather, is represented by the ANN model's structure and parameters. The network's fitness can be determined using an appropriate measure e.g. smallest root mean square prediction error. In complex applications, the number of input variables can be quite large and this problem is further exacerbated in time series studies, where appropriate lags must also be chosen. Therefore, analytical techniques for determining the optimal subset of inputs present the modeller with a distinct advantage.

Maier and Dandy (2000b) reviewed 43 papers on the application of ANNs for hydrological modelling, and found that in many cases the lack of a methodology to determine the input variables raised doubt about the optimality of the inputs obtained. In some instances, inputs were chosen arbitrarily. In other cases, *a priori* knowledge was used and when different methods were employed, such as trial-and-error, often the validation data were used as part of the training process. Intuitively, the preferred approach for determining appropriate inputs and lags of inputs, involves a combination of *a priori* knowledge and analytical approaches (Maier and Dandy 1997; Fernando and Jayawardena 1998; Maier et al. 1998).

There are two broad stages in input determination. Firstly, unsupervised input preprocessing (i.e. discarding redundant inputs) and secondly, supervised input selection (i.e. using the ANN's output in an analytical procedure to determine the significant input variables). In unsupervised input preprocessing, the original set of input variables is processed to produce a subset of inputs containing as much information as possible from the original set. This subset of inputs can then be used in a supervised input selection process to determine which combinations of these inputs result in the network of highest fitness.

In this paper, as a first step in preprocessing the input variables, unsupervised techniques (Section 13.2.1) have been used to reduce the dimensionality of the input space. These techniques are the Self-Organizing Map (SOM) and principal component analysis (PCA). They are compared with the commonly employed approach of using *a priori* knowledge of the system to be modelled. Once redundant inputs have been removed, the input subsets obtained from each unsupervised method are further refined using two supervised input selection methods (Section 13.2.2). These are a hybrid genetic algorithm (GA) and ANN (GA-ANN) and a stepwise ANN modelling procedure. The input determination methods have been used for selecting the optimal subset of input variables for forecasting the concentration of *Anabaena* spp. in the River Murray at Morgan, 4 weeks in advance. The model inputs obtained using each method are compared and each of the six sets of inputs have been used to develop ANN models for forecasting *Anabaena* spp. in the River Murray at Morgan. The ANNs' performance on an independent validation set was used to assess the adequacy of each method of input determination.

13.2 Methods

13.2.1 Unsupervised Input Pre-Processing

A priori identification

In a typical ANN forecasting application, the modeller collects all time series data, subject to availability, that is likely to have an influence on the output variable. Obviously, some knowledge of the system is assumed in determining this set of candidate input variables. However, the data set, although comprehensive, is likely to contain some redundant information. An unsupervised approach to reduce the dimensionality of the input data is to use expert knowledge of the system being modelled. In this way the set of all variables likely to influence the output variable can be reduced to a subset of only those variables most likely to have a significant influence. Expert knowledge can also be used to select the maximum lag of each variable chosen. To aid in this task, it is possible to make use of time series plots of each potential input variable and the output variable. Inspecting the data plots gives a visual indication of any potential relationship that may exist between the input and output variable.

A priori identification is widely used in many ANN applications and since it is dependent on an expert's knowledge, it is very subjective and case dependent. That is why the two analytical procedures (the SOM and PCA) are also being considered.

Self-Organizing Map (SOM)

The Self-Organizing Map was developed by Kohonen (1982) and arose from attempts to model the topographically organised maps found in the cortices of the more developed animal brains. The underlying basis behind the development of the SOM was that topologically correct maps can be formed in an *n*-dimensional array of processing elements (PEs) that did not have this initial ordering to begin with. In this way, input stimuli, which may have many dimensions, can come to be represented by a one- or two-dimensional vector which preserves the order of the higher dimensional data (NeuralWare 1998).

The SOM employs a type of learning commonly referred to as competitive, unsupervised or self-organizing, in which adjacent cells within the network are able to interact and develop adaptively into detectors of a specific input pattern (Kohonen 1990). The SOM can be considered to be as *"neural"* because results have indicated that the adaptive processes utilized in the SOM may be similar to the processes at work within the brain (Kohonen 1990).

The SOM has potential extending beyond its original purpose of modeling biological phenomena. Sorting items into categories of similar objects is a challenging, yet frequent task. The SOM achieves this task by nonlinearly projecting the data onto a lower dimensional display and by clustering these data. This attribute has been used in a wide number of applications ranging from engineering (including image and signal processing and recognition, telecommunications, process monitoring and control, and robotics) to natural sciences, medicine, humanities, economics and mathematics (Kaski et al. 1998).

The Self-Organizing Map Algorithm

In competitive learning, neurons in the network adapt gradually to become sensitive to different input categories. The SOM network generally consists of two layers, an input layer and a Kohonen layer. The input layer is fully connected to the Kohonen layer, which in most common applications is two-dimensional. None of the PEs in the Kohonen layer are connected to each other. The PEs in the Kohonen layer measure the distance of their weights to the input pattern. During the recall phase, the Kohonen PE with the minimum distance is the winner and has an output of 1.0, whilst the other Kohonen PEs have an output of 0.0.

The procedure for determining the winning PE is as follows:

The first step is to determine the extent to which the weights of each PE match the corresponding input pattern. If the input data have N values and are denoted by, $X = (x_i;\ i = 1,...,\ N) \in \Re^n$, then each of the M PEs in the Kohonen layer will also have N weight values and can be denoted by, $W_{ji} = (w_{ji};\ j = 1,...,\ M;\ i = 1,...,\ N) \in \Re^n$. For each of the M Kohonen PEs, the distance, such as the Euclidean distance, is calculated using

$$D_j = \left\| X - W_j \right\| = \left[\sum_{i=1}^{N} \left(x_i - w_{ji} \right)^2 \right]^{\frac{1}{2}}, \qquad j = 1,...,\ M. \tag{13.2}$$

The PE with the lowest value of D_j is the winner during recall. During training, a conscience mechanism adjusts the distances to encourage PEs that are not

winning with an average frequency and to negatively adjust PEs that are winning at an above average frequency. This mechanism ensures that a uniform data distribution develops in the Kohonen layer. In adjusting the distance, a bias, B_j, is added to the distance and forms the new adjusted distance, $D_j^{'}$. The bias is calculated using

$$B_j = \gamma\left(M \times (F_j - 1)\right) \tag{13.3}$$

where γ is a learning coefficient; F_j is the frequency at which the PE j has historically won; and M is the number of PEs in the Kohonen layer. Once B_j and D_j are computed, the adjusted distance, $D_j^{'}$ can be calculated using

$$D_j^{'} = D_j + B_j \tag{13.4}$$

To ensure biological plausibility, lateral interaction with neighbouring PEs is enforced by applying arbitrary network structures called neighbourhood sets, N_c. Throughout the process, all PEs within the winner's neighbourhood set will have their weights updated, whilst PEs outside of this set are left intact. The width or radius of N_c can be time variable. The updating process to implement this procedure is given by

$$W_j(t+1) = \begin{cases} W_j(t) + \alpha(t)(X(t) - W_j(t)) & \text{if } j \in N_c(t) \\ W_j(t) & \text{if } j \notin N_c(t) \end{cases} \tag{13.5}$$

where α is a scalar valued adaptation gain $0<\alpha(t)<1$ and N_c is the neighbourhood set. After the weights have been updated, the next input is presented to the network and the process continues until convergence has been reached. After successively presenting different inputs to the SOM, the net effect is that the weights reflect the topological relationship that exists within the input data (Islam and Kothari 2000).

Implementation of the SOM

The SOM has been used in ecological modelling applications to order data by similarity (e.g. Chon et al. 1996; Foody 1999). In this paper, the SOM is used to cluster the input variables into groups of similar inputs. By then sampling one input from each cluster, it is possible to remove highly correlated, redundant variables from the original data set. The SOM is implemented using the *NCS NeuFrame* software. To cluster the data, the input variables are presented to the network as the SOM's inputs. The software default parameters are used for the learning rate, neighbourhood size and number of epochs. The output of the SOM is obtained using a Dynamic Patterns grid, which shows a dynamic representation of the nodes that are winning each pattern. Each individual cell in the grid represents a node in the Kohonen layer. There is no theoretical principle for

determining the optimum size of the Kohonen layer (Cai et al. 1994), hence, the Kohonen layer was kept large enough to ensure that the maximum number of clusters were formed from the training data. Once the clusters are formed, one input from each cluster is sampled and used in the final subset of input variables.

Principal Component Analysis (PCA)
When many potential variables are available, PCA can be used to reduce the dimensionality of the input data set. By using principal components (PCs), the variables can be transformed to a new, smaller set of variables, which capture most of the information in the original data set. This is achieved by computing factors (new variables) as linear combinations of the old variables. The weights are selected in such a way as to ensure that some optimality criterion is maximised (Masters 1995).

To commence the procedure, a single linear combination of all variables is sought such that the majority of the variation in the training set is captured. After a single dominant factor is found, it is then necessary to find a second factor that captures the remaining information not explained by the first factor. The second PC is chosen such that it is orthogonal to the first. Next a third factor is sought that best captures the remaining information and is orthogonal to the first and second. The process continues until all the variance in the data set is accounted for. If there are p input variables of interest, it is hoped that m, where $m<<p$, different PCs will account for most of the variation in x. As the interrelations among the variables increases, the proportion of variance explained by the first few components increases. Hence, it is common for most of the important information to be concentrated in the first few principal components, with the system noise falling mostly in later components that can be discarded (Masters 1995). However, it is possible, but usually unlikely, that by discarding the later components, some important information may be lost (Jolliffe 1986; Masters 1995). An added advantage of PCA is that each of the computed factors are independent of each other and there is no redundancy in the information that they contain (Masters 1995).

If the input variables have different units, it may be necessary to normalise the data. PCA attempts to capture variation. Numerically, a variation in flow between 10,000 and 20,000 ML/day is much greater than a variation in river level between 1.0 and 5.0 m. However, the effect of each of these variables on the system being investigated may be rather similar and the information content of the flow data is not inherently greater. Hence, by normalising the data, all variables are put on an equal basis in the analysis.

13.2.2 Supervised Input Determination

It is important to note that substantial variation amongst the input variables does not necessarily imply any relationship with the output variable. Hence, after the input dimensionality has been reduced using the unsupervised procedures (*a priori*

identification, SOM and PCA), supervised procedures must ultimately decide upon which variables have the most significant impact on the ANN's forecasting ability.

Genetic Algorithm (GA) Selection of Inputs

A genetic algorithm is a powerful optimisation technique inspired by the principles of natural evolution and selection (Goldberg 1989). Evolutionary algorithms have been widely used in optimising water resources variables (e.g. Simpson et al. 1994; Dandy et al. 1996) and in ecological modelling applications (e.g. Howard and D'Angelo 1995; Downing 1998).

To initiate the technique, a population of random solutions is generated. The fitness of each member of the population can then be evaluated using an objective function and the next generation is produced from the previous one using a process of selection, crossover and mutation.

A GA can be used to select an appropriate combination of inputs to an ANN model. GAs are well suited to this task as they have the ability to search through large numbers of combinations where there may be interdependencies between variables. For the problem of determining inputs to an ANN model, a population of randomly selected ANNs is generated, each with a different subset of input variables as depicted by a binary string. The models are trained and then the output of each ANN is used to determine the predictive error, or fitness of the solution. In this research, the root mean square error (RMSE) between the actual and predicted values is used to determine the fitness of the model. Based on the fitness of each member in the population, a selection scheme can then be employed to create a new population for the next generation. In this way, fit solutions are allowed to live and subsequently breed in the process of crossover. In the crossover process, two parent strings are cut and part of the strings exchanged to produce two new individuals. To maintain genetic diversity and ensure that no important genetic material is overlooked, a mutation operation introduces small random changes.

The process is continued in an iterative manner until the error from the ANN model converges to an acceptable value or the maximum number of generations is completed. Due to the selective pressure applied over the generations, the overall trend is the evolution of higher fitness chromosomes representing optimal input subsets.

Implementation of the GA-ANN

The commercially available software package, *NeuroGenetic Optimizer (NGO)* (BioComp Systems 1998) was used in this research to implement the GA-ANN. The *NGO* uses GAs to evolve ANN structures while simultaneously searching for significant input variables.

Stepwise ANN Modelling Procedure

A stepwise modelling procedure was also used in this study to determine the ANN model inputs (see Masters 1993; Maier et al. 1998). This method involves developing N-bivariate models, where N is the number of input variables. The

input variable that gives the smallest error (e.g. RMSE) is then included in the model. Subsequently, *N-1* models are developed by combining the variable that resulted in the best forecast with each of the remaining variables. This procedure can then be repeated using models with three input variables, four input variables etc., until the addition of any extra variables does not improve model performance. The disadvantages of this procedure are that it is computationally intensive and the synergistic effect of certain combinations of variables may be overlooked.

13.3 Case Study

The ANN models were developed to forecast (4 weeks in advance) a particular species group of cyanobacteria (*Anabaena* spp.) in the River Murray at Morgan, South Australia. In this research, the available data include weekly values of concentrations of the cyanobacterium *Anabaena* spp., total phosphorus, soluble phosphorus, total kjedahl nitrogen (TKN) and silica as well as turbidity, colour, pH, temperature, river levels at Morgan and weekly flows at Lock 7 (for locations see Maier et al. (2000)). All data were available from 1980/1981 to 1995/1996. A full description of the case study is provided in Maier et al. (1998) and Maier et al. (2000). Three additional variables were used in this study, namely, pH, silica concentrations and river levels. pH was included as pH variations can alter phytoplankton community composition i.e. low pH (< 6.0) favours eukaryotes, and high pH (> 8.0) favours cyanobacteria. Silica itself is not a direct requirement for cyanobacterial growth, however, silica is an important nutrient for the growth of diatoms and therefore, it is a key nutrient that determines phytoplankton succession. River levels were also included as they are highly correlated to flow data but exhibit less noise.

13.4 Model Development

Backpropagation networks were developed using the commercially available software package *NeuroGenetic Optimizer (NGO)* (BioComp Systems 1998). The *NGO* evolves ANN structures whilst simultaneously searching for combinations of significant input variables. The following features can be optimised by the *NGO*:

1. the inputs used,
2. the number of hidden layers,
3. the number of hidden layer neurons in each layer, and
4. the transfer functions at each of the hidden and output layer nodes (logistic, hyperbolic tangent or linear).

It has been shown that only one hidden layer is required to approximate any continuous function, given that sufficient degrees of freedom (i.e. connection weights) are provided (Cybenko 1989). Hence, one hidden layer was utilised in this study and this feature was not varied. The number of hidden layer neurons by layer was optimised by the *NGO* and the maximum limit was set at 64 neurons. The test set data were used to choose the optimal network architecture and inputs. The GA-ANN process was conducted in an iterative manner until the maximum number of generations was exceeded. With a population size of 30 networks, it was found that optimal ANN models could be determined within 10 generations.

The *NGO* also has the capability to lock all inputs active and only optimise the network architecture, which means that it can be used to perform the stepwise modelling technique. Unless stated otherwise, the default software parameters were used since the focus is on evaluating the input determination techniques rather than studying the effect of varying the network's parameters.

13.4.1 Performance Measures and Model Validation

The onset, peak and duration of a bloom or growth event are the three most important characteristics describing the occurrence of *Anabaena* spp. The RMSE is not an ideal measure of fitness but it was considered the most suitable error measure, as it places greater emphasis on larger forecasting errors. Even if the RMSEs of several forecasts are similar, the usefulness of the forecasts may differ. For example, two forecasts may have the same RMSE, but one may forecast increases that lead the actual event while the other may lag it, making the former more useful. Therefore, a visual inspection of the plots of actual and predicted results is important in addition to calculating the RMSE between them. In this research, the plots of the actual and predicted (4 week forecast) values were inspected, and the RMSE between them calculated for an arbitrary two-year validation period spanning November 1992 to November 1994.

13.4.2 Data Division

In this paper, the main objective is to compare different input determination techniques. To provide a fair comparison between the different models, it is important that all other modelling factors are held constant and that the models are tested and validated on data that is statistically representative of the data used in the training process. This provides the most rigorous test of a model's performance based on the input selection method since other sources of poor performance such as attempting to validate the model on data outside the range used in training are effectively eliminated. In addition, the ASCE Task Committee on Application of Artificial Neural Networks in Hydrology (2000) observed that "an optimal data set for training would be one that fully represents

the modeling domain and has the minimum number of data pairs in training". To achieve this, the data were divided into training and testing sets using a SOM. The validation data were combined with the remaining data and clustered using the SOM. Once the clusters were formed, two data records from each cluster containing validation data were sampled (i.e. one for each of the training and testing sets). In the instance that a cluster only contained one record other than the validation record, then this record was placed in the training set. The advantage of using the SOM data division technique is that it employs the minimum number of samples for compiling training, testing and validation sets that are statistically similar. When computationally intensive supervised input determination techniques such as the GA-ANN are used, compact training, testing and validation sets are advantageous as they increase processing speed. Since the SOM data division technique allows training, testing and validation sets to be selected that are statistically representative of the same population, a fair comparison of the input selection techniques can be made whilst providing the most rigorous test of each method.

A full description of the use of the SOM for dividing data into statistically similar subsets is provided in Bowden et al. (2000).

13.4.3 Determination of Model Inputs

Six ANN models were developed, each using a different combination of unsupervised and supervised input determination methods (Figure 13.1). After applying the unsupervised techniques to the initial data set, the inputs were reduced to the subsets displayed in Table 13.1. The supervised input determination techniques were performed using the subsets shown in Table 13.1.

13.5 Results and Discussion

Details of each of the models developed are shown in Table 13.2. It can be seen that a variety of different inputs have been selected in each of the models. In general, the models that utilise a GA-ANN for the supervised input determination included a wider variety of input variables than the models developed using the stepwise modelling procedure. The stepwise modelling procedure tended to produce more compact models since the process is terminated once the addition of any extra variables fails to improve the model performance.

GAs are stochastic processes and as such repetition of experimental treatments was performed to evaluate the effect of stochasticity on the set of inputs identified for each GA-ANN model. It was found that by performing multiple runs for each GA-ANN, there were only slight differences in the optimal subset of inputs identified. However, most importantly, it was found that each of the models

identified by the GA-ANN consistently outperformed the models developed by the stepwise ANN modelling procedure when measured on the validation set. Graphical comparisons of the 4-week forecasts of the concentration of *Anabaena* spp. in the River Murray at Morgan for the validation period obtained using model 1 and model 2 are shown in Figures 13.2 and 13.3, respectively. Both of these models were developed using *a priori* knowledge as the unsupervised input processing technique. Model 1 used a hybrid GA-ANN for the supervised input selection, whereas model 2 used the stepwise ANN modelling procedure. The precision of the cell count data can range from ±20% to ±70% depending on the number of *Anabaena* trichomes counted (M.D. Burch, personal communication). Hence, error bars were conservatively set at ±30% and included in Figures 13.2, 13.3 and 13.4. From Figure 13.2, it can be seen that model 1 is able to predict the onset and duration of the two large growth events but underestimates the

Table 13.1. Input Subsets Identified Using the Unsupervised Input Determination Techniques

a priori identification	No. of inputs	PCA	No. of inputs	SOM	No. of inputs
Anabaena lags 1, 2, 3, 4	4	*Anabaena* PCs 1, 2, …, 7	7	*Anabaena* lags 1, 5, 6, 9, 12, 13, 18, 20, 23, 24, 25	11
Turb. lags 1, 2, …, 10	10	Turb. PCs 1, 2, 3, 4	4	Turb. lags 1, 14, 20	3
Col. lags 1, 2, …, 10	10	Col. PCs 1, 2, 3	3	Col. lags 1, 20	2
Temp. lags 1, 2, 3, 4	4	Temp. PCs 1, 2	2	Temp. lags 1, 13	2
Flow lags 1, 2, …, 10	10	Flow PCs 1, 2, 3	3	Flow lags 1, 3, 5, 7, 8, 10, 11, 14, 15, 17, 18, 20, 21, 22, 23, 24, 25	17
pH lags 1, 2	2	pH PCs 1, 2, 3, 4	4	pH lag 1	1
Total. P. lags 1, 2, …, 10	10	Silica PCs 1, 2, 3, 4	4	Sol. P. lag 1	1
Sol. P. lags 1, 2, …, 10	10	Total. P. PCs 1, 2, 3, 4, 5	5	TKN lag 1	1
TKN lags 1, 2, …, 10	10	Sol. P. PCs 1, 2, …, 6	6	River Lev. Lag 1	1
		TKN PCs 1, 2, …, 7	7		
		River Lev. PCs 1, 2, 3	3		
Total No. of Inputs	70		48		39

PC, principal component.

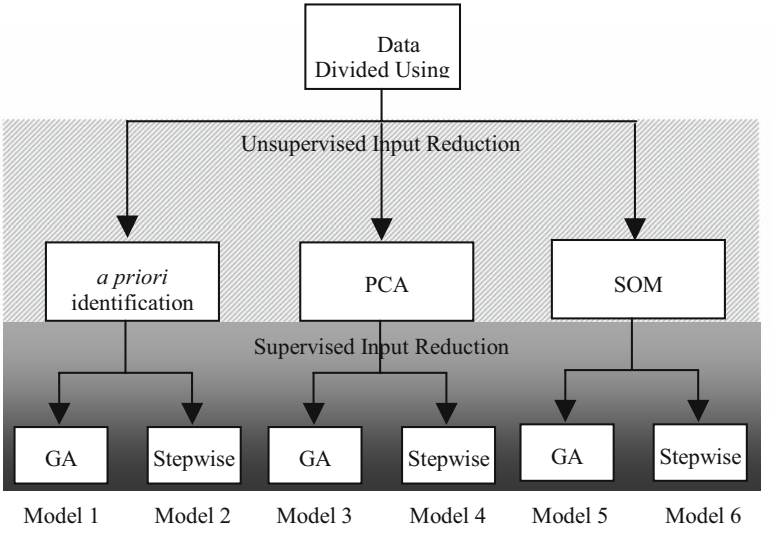


Figure 13.1. The input determination techniques utilised to determine the inputs for each of the six models.

magnitude of the second large peak. In Figure 13.3, it can be seen that model 2 was unsuccessful at predicting the onset of the two large growth events and instead provided a forecast which led the actual events. However, model 2 was successful at predicting the duration and relative magnitude of the two growth events. Model 2 only used flow, lags 1-10 weeks and turbidity, lags 1-10 weeks as input variables, whereas Model 1 made use of 7 different variables including past lags of flow, turbidity, *Anabaena* spp., nutrient data, colour, temperature and pH (Table 13.2). Therefore, other variables are required in addition to flow and turbidity, in order to forecast the onset of the *Anabaena* spp. growth events. It is also important to note that the significant lags selected by the GA-ANN were not intuitive. The superior performance shown by model 1 on the validation set may have resulted from the GA-ANN finding a synergistic combination of input variables and lags. Such a combination of inputs would be impossible to determine using the stepwise ANN modelling procedure. The stepwise ANN procedure is more likely to overlook combinations of interdependent variables, which may together carry significant information.

A plot of the 4-week forecasts for the validation period obtained using model 3 is shown in Figure 13.4. Model 3 used PCA as the unsupervised processing technique and a GA-ANN for the supervised input selection. From Figure 13.4, it can be seen that model 3 picked up the general shape of the two large peaks of *Anabaena* spp. but underestimated the duration and relative magnitude of the second peak. In general, model 3 was unable to perform as well on the validation data as model 1 (the model developed using *a priori* knowledge and the GA-ANN) (Figure 13.2).

Table 13.2. Details of the Models Developed Using Each Combination of Input Determination Technique

Unsupervised method	Supervised Method	Model No.	Inputs Selected	Architecture (N^I - N^H - N^O)	Hidden layer nodes	Output layer node
a priori identification	GA-ANN	1	*Anabaena* lags 1, 3, 4 Turb. lags 5, 6 Col. lags 1, 2, 5, 7, 9 Temp. lags 2, 3, 4 Flow lags 1, 4, 6, 8, 9 pH lags 1, 2 Total P. lags 2, 3, 5, 7 Sol. P. lags 1, 2, 3, 4, 5, 7, 8, 9 TKN lags 2, 3, 5, 6, 7, 8	38 - 57 - 1	28 Logistic 22 Tanh 7 Linear	Tanh
	Stepwise	2	Flow lags 1, …, 10 Turb. lags 1, …, 10	20 - 35 - 1	15 Logistic 11 Tanh 9 Linear	Logistic
PCA	GA-ANN	3	Temp. PC 1 Col. PCs 1, 2 pH PC 1 Silica PCs 1, 3, 4 TKN PCs 1, 5, 6 Total P. PC 1 Sol. P. PCs 2, 3, 4, 5, 6 River Lev. PCs 1, 2 *Anabaena* PCs 1, 3, 4, 6	22 - 36 - 1	16 Logistic 1 Tanh 19 Linear	Tanh
	Stepwise	4	Flow PCs 1, 2, 3 Total P. PCs 1, …, 5 Temp. PCs 1, 2	10 - 52 - 1	6 Tanh 46 Linear	Linear
					20 Logistic 20 Tanh 20 Linear	Logistic
	Stepwise	SOM	GA-ANN	5	*Anabaena* lags 1, 5, 7, 12, 13, 20, 23 Flow lags 7, 8, 10, 13, 15, 17, 20, 22, 25 Turb. lag 1 Col. lags 1, 16 Silica lag 1	20 - 61 - 1

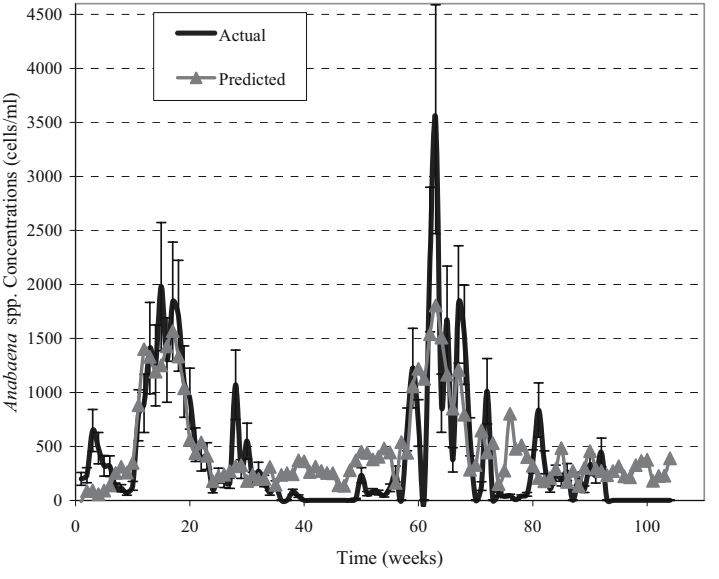


Figure 13.2. Forecasts and actual concentrations of *Anabaena* spp. for the validation period for model 1 (input determination: *a priori* + GA-ANN).

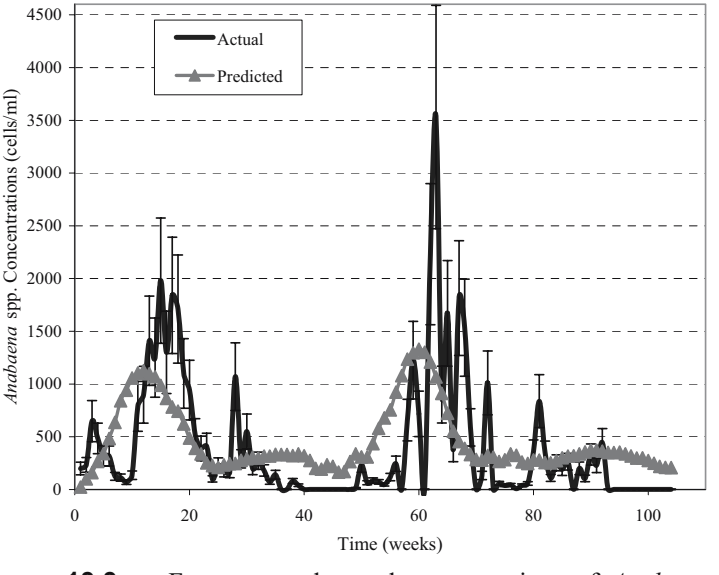


Figure 13.3. Forecasts and actual concentrations of *Anabaena* spp. for the validation period for model 2 (input determination: *a priori* + Stepwise ANN modelling procedure).

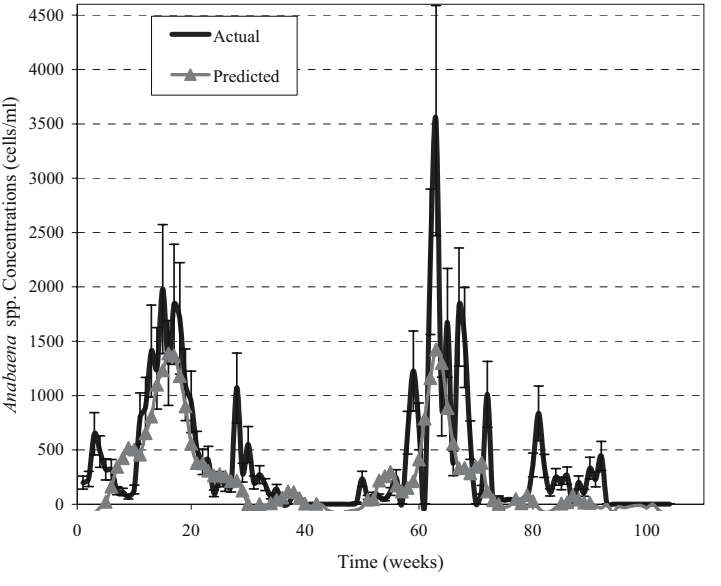


Figure 13.4. Forecasts and actual concentrations of *Anabaena* spp. for the validation period for model 3 (input determination: PCA + GA-ANN).

Table 13.3. RMSE for the 4-week Forecasts

Data Set	*a priori* identification		PCA		SOM	
	GA	Stepwise	GA	Stepwise	GA	Stepwise
Model No.	1	2	3	4	5	6
Training Set	256.3	382.0	371.6	439.8	420.3	398.3
Testing Set	505.4	491.3	561.3	577.9	504.0	502.6
Validation Set	386.2	517.1	436.4	549.4	436.4	602.1

RMSE are given in cells/ml

The RMSEs of the 4-week forecasts for each of the models developed are shown in Table 13.3. Looking at the performance on the validation set, it can be seen that the models developed using *a priori* knowledge performed significantly better than the models developed using PCA and the SOM as unsupervised input processing techniques. This suggests that, where available, expert knowledge on the system being modelled provides a suitable means for removing redundant

input variables and lags of variables. However, this technique is very subjective and dependent on the case study under investigation. In addition, this type of expert knowledge is often unavailable and the only alternative is to proceed with an analytical technique. Based on the performance on the validation set, PCA and the SOM technique both provided an equally suitable means of reducing the dimensionality of the input set. However, the results for the test set suggest that the SOM ANN models perform slightly better than the PCA ANN models. In this theoretical study, the performance of each model has been based on the independent validation set, however, in a real-world application, the decision of which input determination method to use would be based entirely on the test set performance. Thus, this contradictory result requires further investigation but may in part be due to the training, testing and validation sets not being entirely representative of the same population, despite the use of the SOM data division technique. Due to the nature of the *Anabaena* spp. data, it is difficult to compile training, testing and validation sets that are statistically similar.

The results in Table 13.3 also confirm that the GA-ANN performed better than the stepwise ANN procedure when measured on the validation set. However, the GA-ANN and stepwise ANN procedure produced very similar results when measured on the test set. Table 13.2 shows that each of the six models developed contained different network architectures. The network architecture of each model was optimised using the *NGO* and was found to be very dependent on the input subset used. The GA-ANN technique usually identified more input variables than the stepwise ANN procedure, resulting in larger models.

13.6 Conclusions

The results of this study show that the combination of *a priori* knowledge and a hybrid GA-ANN provided the most effective means of identifying the significant input variables. The resulting model (model 1) was able to forecast the onset and duration of the major incidence of *Anabaena* spp. in the validation set with a good level of accuracy. The inputs found to be important in this model include nutrient levels (total phosphorus, soluble phosphorus and TKN) as well as turbidity, colour, temperature, flow, pH and previous concentrations of *Anabaena* (Table 13.2).

Of the analytical unsupervised techniques, the PCA ANN models and the SOM ANN models had identical performance when measured on the independent validation set. Both unsupervised methods provide a suitable means of reducing input dimensionality.

The models (1, 3 and 5) which utilised the GA-ANN as the supervised input determination outperformed the models developed using the stepwise ANN modelling procedure. The GA-ANN had the ability to efficiently evaluate a very large number of networks and consequently, it was able to find synergistic combinations of inputs resulting in superior forecasts for the validation set.

Acknowledgements

Financial support for this research was provided by an Australian Postgraduate Award (Industry) in conjunction with United Utilities Australia Pty Ltd. This support is gratefully acknowledged. The authors would like to thank Mike Burch of the Australian Water Quality Centre for the many helpful discussions, which have been of great benefit to the research.

References

ASCE Task Committee on Application of Artificial Neural Networks in Hydrology (2000) Artificial neural networks in hydrology. II

BioComp Systems, I. (1998) NeuroGenetic Optimizer (NGO). Redmond, WA. Hydrologic applications. Journal of Hydrologic Engineering, ASCE, 5(2): 124-137.

Bowden GJ, Maier HR, Dandy GC (2000) Optimal division of data for neural network models in water resources applications. submitted to Water Resources Research.

Cai S, Toral H, Qiu J, Archer JS (1994) Neural network based objective flow regime identification in air-water two phase flow. The Canadian Journal of Chemical Engineering, 72: 440-445.

Chakraborty K, Mehrotra K, Mohan CK, Ranka S (1992) Forecasting the behavior of multivariate time series using neural networks. Neural Networks, 5: 961-970.

Chon TS, Park YS, Moon KH, Cha EY (1996) Patternizing communities by using an artificial neural network. Ecological Modelling, 90: 69-78.

Cybenko G (1989) Approximation by superpositions of a sigmoidal function. Mathematics of Control Signals and Systems, 2: 203-314.

Dandy GC, Simpson AR, Murphy LJ (1996) An improved genetic algorithm for pipe network optimisation. Water Resources Research, 32(2): 449-458.

Downing K (1998) Using evolutionary computation techniques in environmental modelling. Environmental Modelling & Software, 13(5-6): 519-528.

Fernando DAK, Jayawardena AW (1998) Runoff forecasting using RBF networks with OLS algorithm. Journal of Hydrologic Engineering, 3(3): 203-209.

Foody GM (1999) Applications of the self-organising feature map neural network in community data analysis. Ecological Modelling, 120: 97-107.

Goldberg DE (1989) Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, Reading, MA, 412 pp.

Howard LM, D'Angelo DJ (1995) GA-P: a genetic algorithm and genetic programming hybrid. IEEE Expert, 10(3): 11-15.

Islam S, Kothari R (2000) Artificial neural networks in remote sensing of hydrologic processes. Journal of Hydrologic Engineering, 5(2): 138-144.

Jolliffe IT (1986) Principal Component Analysis, Springer-Verlag New York Inc., New York, 271 pp.

Kaski S, Kangas J, Kohonen T (1998) Bibliography of Self-Organizing Map (SOM) Papers: 1981-1997. Neural Computing Surveys, 1: 102-350.

Kohonen T (1982) Self-organized formation of topologically correct feature maps. Biological Cybernetics, 43: 59-69.

Kohonen T (1990) The Self-Organizing Map. Proc. IEEE, 78(9): 1464-1480.

Lachtermacher G, Fuller JD (1994) Backpropagation in hydrological time series forecasting. In K. W. Hipel, A. I. McLeod, U. S. Panu, and V. P. Singh (Eds), Stochastic and Statistical Methods in Hydrology and Environmental Engineering, Kluwer Academic Publishers, Dordrecht, pp. 229-242.

Maier HR, Dandy GC (1997) Determining inputs for neural network models of multivariate time series. Microcomputers in Civil Engineering, 12(5): 353-368.

Maier HR, Dandy GC (2000a) Application of Artificial Neural Networks to Forecasting of Surface Water Quality Variables: Issues Applications and Challenges. In R. S. Govindaraju and A. R. Rao (Eds), Artificial Neural Networks in Hydrology, Kluwer Academic Publishers pp. 348.

Maier HR, Dandy GC (2000b) Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. Environmental Modelling and Software, 15: 101-124.

Maier HR, Dandy GC, Burch MD (1998) Use of artificial neural networks for modelling cyanobacteria *Anabaena* spp. in the River Murray, South Australia. Ecological Modelling, 105: 257-272.

Maier HR, Sayed T, Lence BJ (2000) Forecasting cyanobacterium *Anabaena* spp. using B-spline neurofuzzy models, 2nd International Conference on Applications of Machine Learning to Ecological Modelling, Adelaide, Australia.

Masters T (1993) Practical Neural Network Recipes in C++, Academic Press, San Diego, 493 pp.

Masters T (1995) Neural, Novel and Hybrid Algorithms for Time Series Prediction, John Wiley and Sons, New York, 514 pp.

NeuralWare (1998) Neural Computing: A Technology Handbook for NeuralWorks Professional II/PLUS and NeuralWorks Explorer, Aspen Technology Inc., USA, 324 pp.

Simpson AR, Dandy GC, Murphy LJ (1994) Genetic algorithms compared to other techniques for pipeline optimisation. Journal of Water Resources Planning and Management, ASCE, 120(4): 423-443.

Chapter 14

Utility of Sensitivity Analysis by Artificial Neural Network Models to Study Patterns of Endemic Fish Species

M. Gevrey · S. Lek · T. Oberdorff

14.1 Introduction

Artificial Neural Networks (ANNs) have become more and more frequently used in ecology in the last decade, essentially to resolve forecasting problems (Cornuet *et al.* 1996; Recknagel *et al*. 1997; Guégan *et al.* 1998; Clair and Ehrman 1998; Ozesmi and Ozesmi 1998; Maier and Dandy 1999; Laberge *et al*. 2000).

Many studies have shown that this tool has a better predictive power than the classical linear methods (Lek *et al.* 1996b; Brey *et al.* 1996; Ramos-Nino *et al*. 1997; Starrett *et al.* 1997; Huntingford and Cox 1997; Paruelo and Tomasel 1997; Whitehead *et al.* 1997). Nevertheless ANNs are known as a "black box" type model, unable to clarify the contribution of the explanatory variables to the dependent one. However, even if the prediction capacity is of prime importance, it is also necessary, particularly in ecology, to evaluate the way explanatory variables contribute to the explanation of the ecological processes involved. Several authors have thus focused on the analysis of output variables sensitivity with respect to the input (Garson 1991; Goh 1995; Lek *et al.* 1996a,b; Maier and Dandy 1996; Balls *et al.* 1996; Scardi 1996; Seginer 1997; Dimopoulos *et al*. 1995; 1999). In this paper, two algorithms developed by Lek (Lek *et al.* 1996a,b) and Dimopoulos (Dimopoulos *et al*. 1995; 1999) are evaluated and compared.

These two methods reinvestigate a previous study analysing patterns of endemic riverine fish species richness at the global scale (Oberdorff *et al*. 1999). In this study, the endemic species richness was directly dependent on species diversity (e.g. total species richness). Total species richness was itself influenced by factors related to components of river size (i.e. surface area of the drainage basin) and to a lesser extent, energy availability (i.e. net primary productivity).

This study comprises four sections corresponding to: i) the development of the two methods, ii) the presentation of the dataset, iii) the investigation of results obtained for predictions and those obtained with the contribution methods, iv) the discussion of the ecological significance of the results obtained.

14.2 Contribution of Environmental Variables

In the Multi-Linear Regression (MLR) model, the influence of each variable can be roughly assessed by checking the final values of the regression coefficients. However, it is more difficult with ANNs to find the contribution of the input variables directly from the models, specific algorithms are necessary to use.

Most authors have used the principle of step by step elimination of the variables to determine this influence (Balls *et al.* 1996; Maier and Dandy 1996) but others have tried different methods using connection weights (Garson 1991; Goh 1995), the perturbation of input variables (Scardi 1996), the partial derivatives of the output according to the input variables (Dimopoulos *et al*. 1995; 1999) or the successive study of the variables by a variation of one of them while the others are fixed to a determined value (Lek *et al.* 1996a,b), etc. The two methods retained for this study are the Profile method (Lek *et al.* 1996a,b) and the PaD method (Dimopoulos *et al*. 1999).

The "Profile algorithm"

This method was suggested by Lek *et al* (1996a, b). The general idea is to study each input variable successively, to do so the others are blocked during the utilisation of the model. The principle of this algorithm is to construct a fictitious matrix considering the range of all input variables. In greater detail, the values of each variable are divided into 12 values at equal intervals between their minimum and maximum values. For all variables except one, the 12 values are set at their minimum values, then successively their first quartile, median, third quartile and maximum. For each studied variable, five values for each of the 12 points are obtained. These five values are reduced to the median value. Then the profile of the output variable can be plotted for 12 values levels of the variable considered. The same calculations can then be repeated for each of the other variables. For each variable a curve is then obtained, which gives a set of profiles of the variation of the dependent variable according to the increase of the input variables.

The "Pad algorithm"

This method gives two different results, the first one being a profile of the output for each input variable and the second being a classification of the variable in increasing order of importance.

"The derivatives profile"

This is a sensitivity analysis proposed by Dimopoulos *et al.* (1995; 1999). It is based on the principle of partial derivatives of the ANN response with respect to each descriptor. When the input x_j is modified, the output y_j changes, $y_j=f(x_j)$. Then the sensitivity of the network outputs according to small input perturbations can be studied, which is represented by the Jacobian matrix $dy/dx^T = \left[\partial y \ / \partial x \ \right]_{m\times n}$. For a network with n inputs, one hidden layer with ni nodes, and one output (i.e. m=1), the gradient vector of y_j with respect to x_j is $d_j = \left[d_{j1}, \ldots, d_{je}, \ldots, d_{jn}\right]^T$ (Dimopoulos *et al*. 1995), with:

$$d_{je} = S_j \sum_{i=1}^{ni} w_{is} I_{ij} \left(I - I_{ij}\right) w_{ei} \tag{14.1}$$

(on the assumption that a logistic sigmoid function is used for the activation. When S_j is the derivative of the output node with respect to its input, I_{ij} is the output node for the input x_j, w_{is} and w_{ei} are the weights between the s^{th} output node and i^{th} hidden node, and between the e^{th} input node and the i^{th} hidden node).

A set of graphs of the partial derivatives versus each corresponding input variable can then be plotted, and enable the direct access of the variation of the input variable influence on the output. One example of an interpretation of these graphs is that, if the partial derivative is negative then, for this value of the studied variable, the output variable will tend to decrease while the input variable will increase. Inversely, if the partial derivatives are positive, the output variable will tend to increase while the input variable will increase. Derivatives equal to zero imply independence between the output and the considered input variable.

"The relative contributions"

The sensitivity of the ANN output for the data set with respect to an input is calculated by a sum of the square partial derivatives values obtained per variable:

$$SSD_e = \sum_{j=1}^{N} (d_{je})^2 \tag{14.2}$$

One SSD (Sensitivity for a Set of Data) value is obtained per input variable. The SSD values allow classification of the variables according to their increasing contribution to the output variable in the model. The input variable, which has the higher SSD value, is the variable mostly influencing the output variable.

14.3 Application to Ecological Data

Dataset:

Data were used from 136 rivers of the Northern Hemisphere located as follows: 58 in Africa (43%), 52 in Europe (38%), 19 in America (14%) and 7 in Asia (5%); 49 rivers between 0° and 10° (36%), 11 between 11° and 20° (8%), 8 between 21° and 30° (6%), 17 between 31° and 40° (12%), 33 between 41° and 50° (24%), 13 between 51° and 60° (10%) and 5 between 61° and 70° (4%). This database is the same as the one used by Oberdorff (1999) with the addition of some new data.

Studied variables retained are as follows: total species richness (TSR), referring to the total number of riverine fish species collected from the entire drainage basin; endemic species richness (ESR), referring to the total number of endemic fish species collected from the entire drainage basin (endemic species are defined as those inhabiting only one drainage basin, narrow endemic); total surface area of the drainage basin (km^2) (SAD); and net primary productivity ($kg^{-2}y^{-1}$) (NPP) (because the net aquatic primary productivity data were not available for rivers in the literature, the formula given by Lieth (1975) was used, which included the

mean annual air temperature and the mean annual rainfall to estimate average values of terrestrial primary productivity).

The prediction method

ANNs were employed to predict the endemic species richness using a set of explanatory variables at the input, which were the total species richness, the surface of the drainage basin and the net primary productivity. Because the endemic species richness depends normally only on total species richness (Oberdorff *et al.* 1999), ANNs were also employed to predict the total species richness and investigate the contribution of the drainage basin surface area and the net primary productivity. There is a range of different types of ANNs but the most widely used is the multi-layered perceptron which is trained using the algorithm of backpropagation of errors (Rumelhart *et al*. 1986). It is based on a supervised learning of a known data matrix, a correction of the connection weights is done in order to obtain a minimal error, that is the method of gradient descent based on the difference between the observed and the expected outgoing signals. The final model obtained can be used to carry out predictions.

For the prediction of ESR, a three layer feed-forward (3-5-1) neural network was used: i.e. 3 input neurones corresponding to each quantitative variable (TSR, SAD, NPP), one hidden layer with 5 neurones determined as the optimal configuration (best compromise between bias and variance, Geman *et al.* 1992; Kohavi 1995) and one output neurone for the endemic species richness. The activation function used is the log sigmoïd function. To test and validate the model, a leave-one-out procedure was investigated (Efron 1983; Jain *et al.* 1987), where each observation is tested using a model trained by all the observations. In reality, each observation (river) is unique and can be added to the training sample according to a posteriori validation. In this way, 136 training phases were performed with 135 observations followed by 136 testing phases with only one observation.

For the prediction of TSR, the network used had two input neurones: SAD and NPP, also 5 hidden neurones and one output neurone for TSR. The above test and validation procedure was used for the model.

The computational program was undertaken using Matlab® software release 5.3 on PC.

14.4 Results

14.4.1 Predictive Power

The results of the ANNs model to predict ESR by the leave-one-out testing procedure, with 500 iterations and 5 neurons in the hidden layer, are presented in

Fig. 14.1. The determination coefficient between observed and estimated values testified the predictive power of the model (r^2=0.92). Figure 1a shows that the ANNs gave satisfactory results over the whole range of the dependent variable values. The points are well aligned on the diagonal of the perfect-fit line. There are many more low values of ESR but the highest are better predicted. In Fig. 14.1b, the results of the study of the relationship between residuals and estimated values shows complete independence (r=-0.018, n=136, P=0.838) and an average of residuals equal to zero (equal distribution of the points on both sides of the x-axis). More low value points can be again seen, which also correspond to the highest residuals deviation. In Fig. 14.1c, the distribution of residuals was compared to a normal distribution but it appears that there is an exaggerated clustering of residuals at zero for the distribution of residuals to be normal. Thus, the assumption of normality may not hold. Lillieffors test of normality of residuals gives a maximum difference of 0.268 (P<0.001).

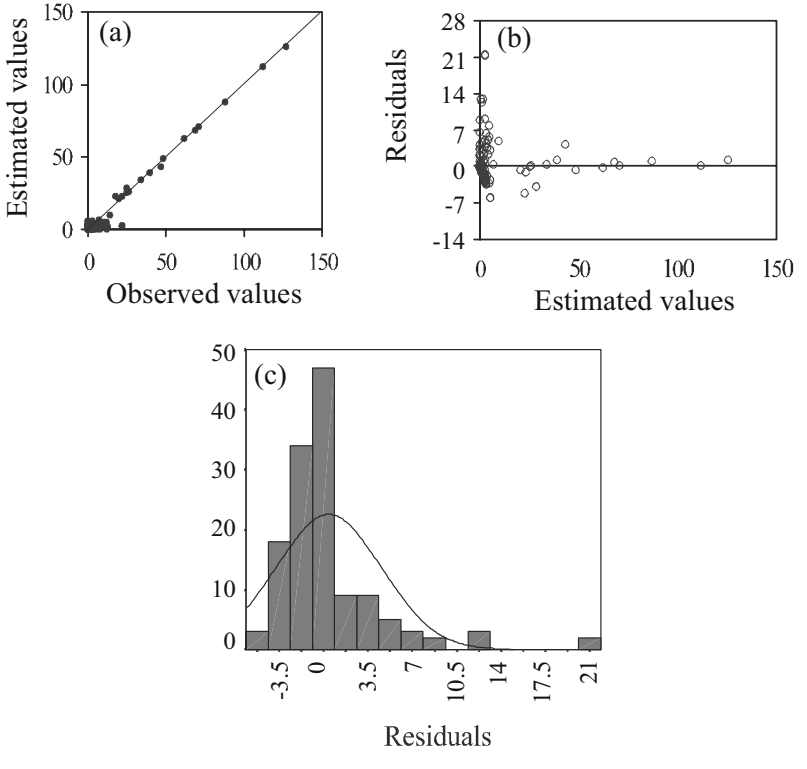


Fig. 14.1. (a) Relationship between observed and estimated values of ESR, (b) Relationship between the residuals and the estimated values of ESR, (c) Distribution of residuals (observed values- estimated values of ESR).

The results of the ANNs model to predict TSR are in Figure 14.2. The determination coefficient is lower than for ESR (r^2=0.86). The points are less well

aligned on the diagonal of the perfect-fit line. (Fig. 14.2a). There are many low values of TSR, which are underestimated, but there is a complete independence between the residuals values and the estimated values (n=136, r= 0.045, P=0.606) (Fig. 14.2b). Nevertheless, there is an exaggerated clustering of residuals at zero in order for the distribution to be normal. The assumption of normality may not hold. Lillieffors test of normality of residuals gives a maximum difference of 0.225 (P<0.001) (Fig. 14.2c).

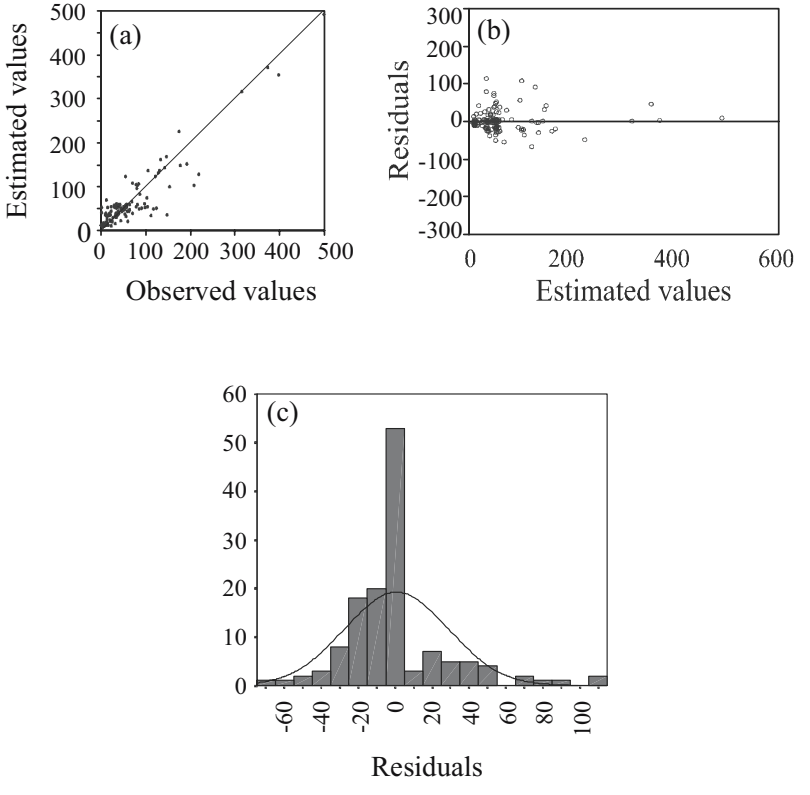


Fig. 14.2. (a) Relationship between observed and estimated values of TSR, (b) Relationship between the residuals and the estimated values of TSR, (c) Distribution of residuals (observed values- estimated values of TSR).

14.4.2 Sensitivity Analysis

Here are given the results of the investigation of the two sensitivity methods.

Profile method

Figure 14.3 shows the results of the contribution analysis with the 3 variables: TSR, SAD and NPP in the predictive model of ESR. The variable TSR has a curve, which follows the entire range of ESR values when the SAD and NPP curves are what can be called "crushed".

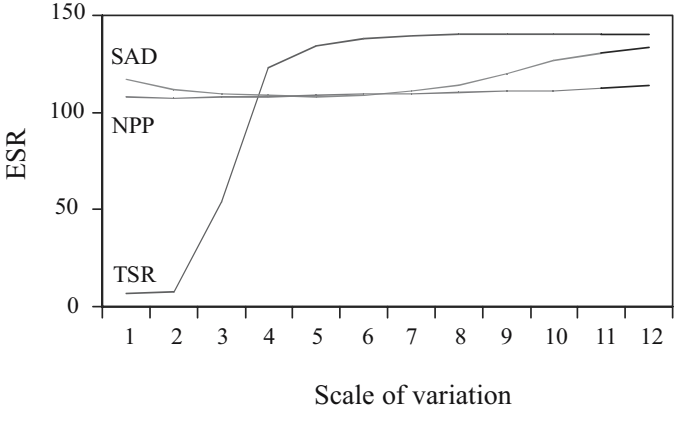


Fig 14.3. Contribution of the three independent variables (TSR, SAD and NPP) used in the 3-5-1 ANN model by the Profile algorithm.

Figure 14.4 is the graph of the contribution analysis with the 2 variables SAD and NPP in the predictive model of TSR. The SAD curve has a positive linear shape. The relationship between NPP and TSR is non linear and positive.

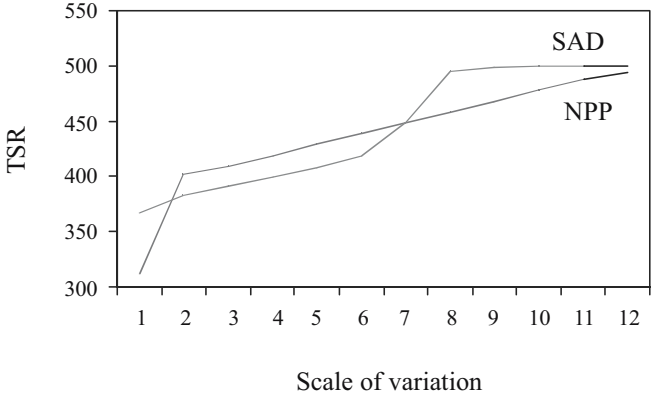


Fig 14.4. Contribution of the two independent variables (SAD and NPP) used in the 2-5-1 ANN model by the Profile algorithm.

"Pad method": The derivatives profile

Fig. 14.5 shows the results of the partial derivative method, that is three graphs (one for each variable), which represent the partial derivative of the output with respect to the input values versus the values of this input.

As it can be seen in Fig. 14.5a, the values of partial derivatives of ESR with respect to TSR are nearly all positive or equal to zero for the whole range of TSR values. The partial derivatives plot can be divided into three parts: (1) TSRs lower than 50: the partial derivatives of ESR are equal to zero, there is no increase in ESR for an increase in TSR; (2) TSRs between 50 and 150: the partial derivatives values of ESR increase. An increase in TSR leads to an increase in ESR; (3) TSRs higher than 150: the values of ESR partial derivatives stay positive but are lower. An increase in TSR leads to a more moderate increase in ESR.

Concerning the plots of the ESR partial derivatives with respect to SAD and NPP, Fig. 14.5b and Fig. 14.5c respectively, both the points are mainly close to zero. There is an independence of the ESR variable versus SAD and NPP.

Fig 14.6 shows the results of the partial derivative of TSR with respect to the two variables SAD and NPP.

Fig. 14.6a is the graph of the TSR partial derivatives with respect to SAD versus SAD. All the partial derivative values are positive. In the case of the low SAD values the partial derivatives values are high, and become lower with the increase in SAD.

Fig. 14.6b shows the graph of the TSR partial derivatives with respect to NPP versus NPP. The partial derivatives plot can be divided into three parts: (1) NPPs lower than 1500 $kg^{-2}y^{-1}$, the partial derivatives are near zero, there is an independence of NPPs for these values; (2) NPPs between 1500 and 2250 $kg^{-2}y^{-1}$, the partial derivatives are positive and high, an increase in NPP leads to an increase in TSR; (3) NPPs higher than 2250 $kg^{-2}y^{-1}$, the partial derivatives of TSR are negative, an increase in NPPs leads to a decrease in TSR.

The relative contributions

The algorithm permitted the calculation of a value of SSD per variable. Several simulations were undertaken to obtain those SSDs. Because there was variation between simulations in the obtained values of SSD and the relative proportions between the SSDs of the 3 variables were conserved, thus the SSDs were expressed as a percentage of the sum of the three SSDs and then averaged. The values obtained are: SSD_{TSR}=76.77%, SSD_{SAD}=19.74% and SSD_{NPP}=3.49%. Therefore, TSR is the most significant variable, followed by SAD and then NPP, which has a very low contribution. The differences between the three SSDs are sufficient for the t-test to be significant ($P<0.01$).

The same method was applied with TSR as the output variable and SAD and NPP as the inputs. The results obtained are: SSD_{SAD}=70.67% and SSD_{NPP}=29.33%. The t-test gives a significant difference between SAD and NPP ($P<0.01$).

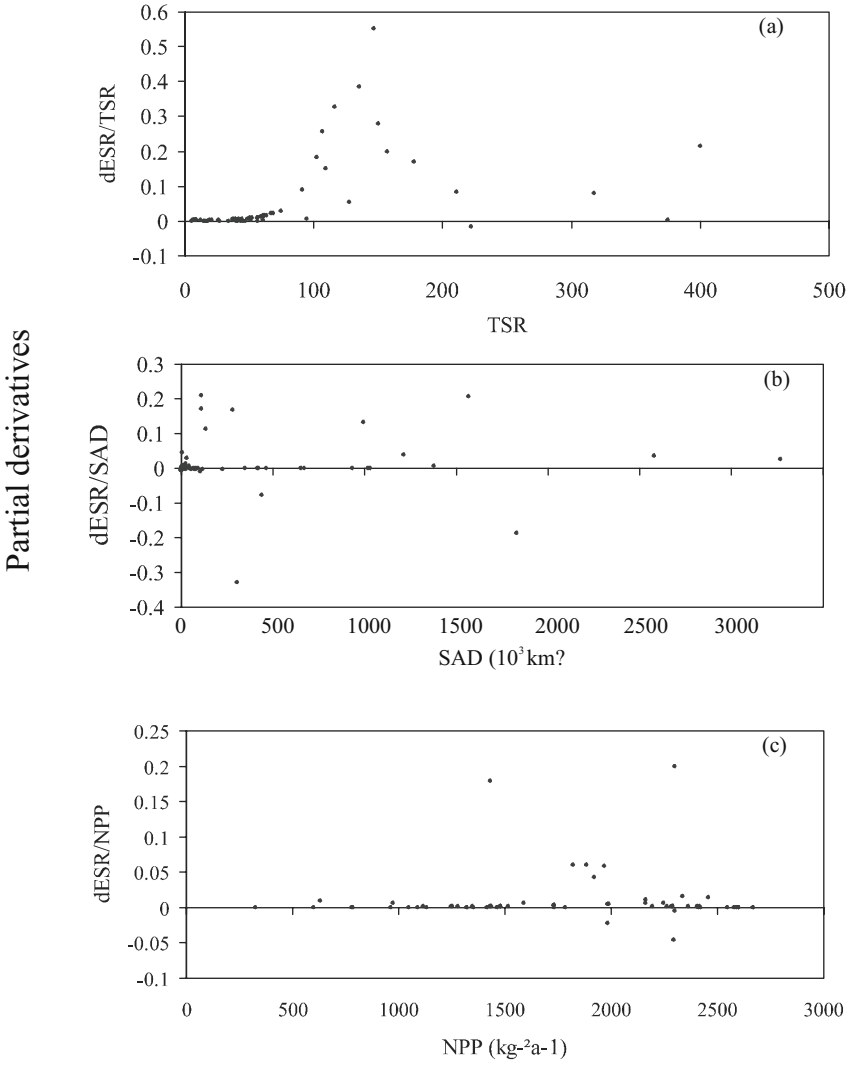


Fig 14.5. Partial derivatives of the ANN model response (ESR) with respect to each independent variables (Pad algorithm, Derivatives Profile) (a) TSR, (b) SAD, (c) NPP.

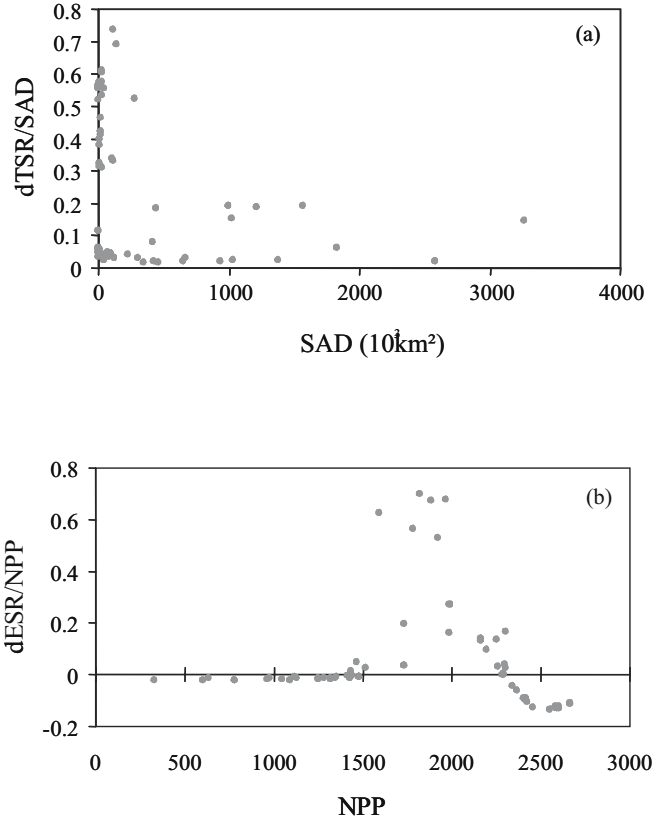


Fig 14.6. Partial derivatives of the ANN model response (TSR) with respect to each independent variables (Pad algorithm, Derivatives Profile) (a) SAD, (b) NPP.

14.5 Discussion

The power of Artificial Neural Networks is verified by a very high determination coefficient between the observed values and the estimated values for both models (r^2=0.92 for ESR prediction and r^2=0.86 for TSR prediction). The results are in agreement with the literature, in which ANN performances have repeatedly been reported to surpass those of more traditional methods such as MLR (*cf.* reference cited in the introduction). This may point to the predominantly non-linear relationships between the studied variables on the one hand, and on the other hand, the ability of ANNs to take directly into accounts any non-linear relationships between the dependent variables and each independent variable (Lek

et al. 1996b). Therefore, ANNs appear to constitute a powerful alternative in predictive ecological modelling.

With both the Profile method and the Pad method, TSR is found to be the most important variable in explaining the variability of the endemic species richness. This is in agreement with the conclusions made by Oberdorff (1999). Moreover, the way that the TSR varies the ESR is also similar in the two methods. For low values of TSR, ESR does not increase, then ESR increases with average values of TSR and with high TSR values, ESR increases but more moderately.

The two variables SAD and NPP are not expressed by both methods. If ESR is predicted only by the TSR the determination coefficient found will be r^2=0.83. With the Profile method their curves have a limited range and with the PaD method their partial derivatives values are almost all near zero. This is in agreement with the study of Oberdorff (1999), where the total species richness is the only variable influencing the endemic species richness.

The Profile and PaD methods were applied to the total species richness to test which of the two variables NPP and SAD is the most important, and how they intervene in the TSR variation. In this case, the curves of SAD and NPP are well expressed with the Profile method, and the partial derivatives differ from zero with the PaD method. TSRs increase with SADs using both methods, but they give different results for NPPs. With the Profile method TSRs increase with NPPs whereas with the Pad method, TSRs increase only for values of NPP between 1500 and 2250 $kg^{-2}y^{-1}$.

In conclusion, these methods show that the endemic species richness is dependent only on the total species richness and that total species richness is dependent on the surface area of the drainage basin and to a lesser extent, the net primary productivity.

Disadvantages

The principal disadvantage of the Profile method is the use of a fictitious matrix. In fact, twelve values were retained for each variable, taken from between their minimum and maximum values at equal intervals. The plotted curves for each variable are then smoothed, and are therefore not precise. For a definite value of one independent variable, it is not directly possible to find the value of the dependent variable. The curves can only demonstrate how the variables influence the output in relation to an increase in its values; it is a general view. If the contribution order of the variable is needed, this method is not able to give it directly. An analysis for a definite value can only be carried out by comparison of the curves' slopes.

Because a graph is obtained for each variable with the Pad method, it is less easy to get a general view of the influence of all the variables, as with the Profile method.

Advantages

The Pad method seems more favorable than the other. This method uses the real values of the observations. Moreover, it permits the direct obtaining of two things; the first being the evolution of the output according to the increase in each

independent variable and this with the precise values of the independent variable. The second is a classification of the independent variable in increasing order of influence. Moreover, when a variable is more dominant than another, the partial derivatives obtained for the other are near to zero.

The profile method, because it presents all the variables at the same scale permits a general view of how the independent variables influence the output. When a variable is dominant, a curve with a large range is represented for this variable while the other curves are "crushed".

These methods are able to provide explanations for the model. The most influential variable is known, as is the order of influence of all the variables, and the way that these variables intervene in the model.

14.6 Conclusions

The results obtained with both methods match closely with the previous results. The predictive power of ANNs has often been demonstrated, and this new study puts to the fore their explicative power which is very interesting in ecological research.

This article paves the way forward for broad research concerning the contribution of the input variables in ANN's, firstly by the use of other databases to test the methods, secondly by the discovery of new methods and finally by the investigation of other existing methods.

References

Balls GR, Palmer-Brown D, Sanders GE (1996) Investigating microclimatic influences on ozone injury in clover (*Trifolium subterraneum*) using artificial neural networks. New Phytol. 132, 271-280

Brey T, Jarre-Teichmann A, Borlich O (1996) Artificial neural network versus multiple linear regression: predicting P/B ratios from empirical data. Marine Ecology Progress Series, 140, 251-256

Clair TA, Ehrman JM (1998) Using neural networks to assess the influence of changing seasonal climates in modifying discharge, dissolved organic carbon, and nitrogen export in eastern Canadian rivers. Water Resources Research, 34(3), 447-455

Cornuet JM, Aulagnier S, Lek S, Franck P, Solignac M (1996) Classifying individuals among intra-specific taxa using microsatellite data and neural networks. C. R. Acad. Sci. Paris, Sciences de la vie 319, 1167-77

Dimopoulos Y, Bourret P, Lek S (1995) Use of some sensitivity criteria for choosing networks with good generalization ability. Neural Process. Lett. 2(6), 1-4

Dimopoulos Y, Chronopoulos J, Chronopoulou-Sereli A, Lek S (1999) Neural network models to study relationships between lead concentration in grasses and permanent urban descriptors in Athens city (Greece). Ecological Modelling 120, 157-165

Efron B (1983) Estimating the error rate of a prediction rule: improvement on cross-validation. J. Am. Stat. Assoc. 78, 316-330

Garson GD (1991) Interpreting neural network connection weights. Artificial Intelligence Expert 6, 47-51

Geman S, Bienenstock E, Doursat R (1992) Neural Network and the bias/variance dilemna. Neural Comput. 4, 1-58

Goh ATC (1995) Back-propagation neural networks for modelling complex systems. Artificial Intelligence Engineering 9, 143-151

Guégan JF, Lek S, Oberdorff T (1998) Energy availability and habitat heterogeneity predict global riverine fish diversity. Nature 391, 382-384

Huntingford C, Cox PM (1997) Use of statistical and neural network techniques to detect how stomatal conductance responds to changes in the local environment. Ecological Modelling 97, 217-246

Jain AK, Dube RC, Chen C (1987) Bootstrap techniques for error estimation. IEEE Trans. Patt. Anal. Mach. Intell. PAMI 9, 628-633

Kohavi R (1995) A study of cross-validation and bootstrap for estimation and model selection. Proceeding of the 14th International Joint Conference on Artificial Intelligence, Morgan Kaufmann Publishers, pp. 1137-1143

Laberge C, Cluis D, Mercier G (2000) Metal bioleaching prediction in continuous processing of municipal sewage with Thiobacillus ferrooxidans using neural networks. Water. Resource Research. 34 (4), 1145-1156

Lek S, Belaud A, Baran P, Dimopoulos I, Delacoste M (1996a) Role of some environmental variables in trout abundance models using neural networks. Aquatic Living. Resource 9, 23-29

Lek S, Delacoste M, Baran P, Dimopoulos I, Lauga J, Aulagnier S (1996b) Application of neural networks to modelling nonlinear relationships in ecology. Ecological Modelling 90, 39-52

Lieth H (1975) Modelling the primary productivity of the world. Primary Productivity of the Biosphere, eds. Lieth, H. & Whittaker, R. H. New York: Springer-Verlag, pp. 237-263

Maier HR, Dandy GC (1996) The use of artificial neural networks for the prediction of water quality parameters. Water Resources Research, 32 (4), 1013-1022

Maier HR, Dandy GC (1999) Empirical comparison of various methods for training feed-forward neural networks for salinity forecasting. Water Resources Research, 35 (8), 2591-9596

Oberdorff T, Lek S, Guégan JF (1999). Patterns of endemism in riverine fish of the Northern Hemisphere. Ecology Letters, 2, 75-81

Özesmi SL, Özesmi U (1999) An artificial neural network approach to spatial habitat modelling with interspecific interaction. Ecological Modelling 116, 15-31

Paruelo JM, Tomasel F (1997) Prediction of functional characteristics of ecosystems: a comparison of artificial neural networks and regression models. Ecological Modelling 98, 173-186

Ramos-Nino ME, Ramirez-Rodriguez CA, Clifford MN, Adams MR (1997) A comparison of quantitative structure-activity relationships for the effect of benzoic and cinnamic acids on Listeria monocytogenes using multiple linear regression, artificial neural network and fuzzy systems. Journal of Applied Microbiology 82, 168-176

Recknagel F, French M, Harkonen P, Yabunaka KI (1997) Artificial neural network approach for modelling and prediction of algal blooms. Ecological Modelling 96, 11-28

Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating error. Nature, 323, 533-536

Seginer I (1997) Some artificial neural network applications to greenhouse environmental control. Computers and Electronics in Agriculture 18, 167-186

Starrett SK, Starrett SK, Adams GL (1997) Using artificial neural networks and regression to predict percentage of applied nitrogen leached under turfgrass. Commun. Soil Sci. Plant anal., 28, 497-507

Whitehead PG, Howard A, Arulmani C (1997) Modelling algal growth and transport in rivers: a comparison of time series analysis, dynamic mass balance and neural network techniques. Hydrobiologia 349, 39-46

Part IV

Prediction and Elucidation of Lake and Marine Ecosystems

Chapter 15

A Comparison between Neural Network Based and Multiple Regression Models for Chlorophyll-*a* Estimation

C. Karul · S. Soyupak

15.1 Introduction

15.1.1 Eutrophication in Water Bodies and Relevant Models

Eutrophication and associated algal blooms are serious problems in many lakes and reservoirs. Deterioration of water quality for human consumption, limitation of recreational use, depletion of dissolved oxygen levels below tolerable levels for certain fish species and severe ecosystem degradation are amongst the adverse effects of eutrophication (Ryding and Rast 1989).

Various types of simulation models have been developed to predict the magnitude and timing of algal blooms. One major class of models for predicting water quality parameters (i.e. algal concentrations) can be named "as empirical water body models". They were primarily developed as extensions of the phosphorus model. Models developed by Dillon and Rigler (1974); Rast and Lee (1978); and Bartsch and Gakstatter (1978) are based on a fit of a log-log plots of Chlorophyll-a and P. They are the typical examples of such models. Later Smith and Shapiro (1981) have presented a modified correlation that takes into account the potential nitrogen limitation. Specifically, some regression models predict the algal concentrations as a function of flushing corrected average annual phosphorus inflow concentrations as summarized by Ryding and Rast (1989). Some empirical statistical models developed for 233 Florida lakes predict logarithm of chlorophyll-a concentrations as functions of "logarithm of total phosphorus concentrations" and "logarithm of total nitrogen concentrations"(Canfield et al. 1983; 1984). The authors has stressed the applicability of simple models with very few parameters for predicting chlorophyll-a concentrations and they further

emphasize the importance of aquatic macrophytes. The final outcome of their study was that the nitrogen was limiting nutrient for hypertrophic lakes.

Deterministic models (Jorgensen 1976; Benndorf and Recknagel 1982), time series analysis models (Whitehead and Hornberger 1984), and fuzzy-logic models (Recknagel et al. 1994) are the other major classes of models for the same purpose.

Since the empirical models utilizes limited number of parameters to predict chlorophyll-a concentrations, it is natural to conclude that they provide rough estimates with low degree of precision with serious oversimplifications. The processes that lead to eutrophication of water bodies are known to involve extremely complex behaviors with nonlinear relations between system parameters and the system responses that can not be readily explained with simplistic approaches. Starting from this idea, the authors of this manuscript thought that developing artificial neural network tools that are adequately trained with several environmental factors could be a better approach for more precise predictions. The characteristics of artificial network methodology allow learning complex systems and predicting their responses with high degree of precision if adequately applied.

15.1.2 Artificial Neural Networks

The use of artificial neural networks in solving complex problems is becoming popular in many disciplines due to their capability to 'learn' non-linear relations. The method of artificial neural network has been inspired by biological nervous system. Neural Network, in computer science, is highly interconnected network of information-processing elements that mimics the connectivity and functioning of the human brain. One of the most significant superiority of artificial neural networks is their ability to learn from a limited set of examples. Artificial neural networks are being created mimicking the structure and functioning of biological neural networks, in an artificial way.

A properly trained and verified artificial neural network for the specific problem of interest recognizes the data and makes predictions with desirable accuracy. The problem of interest can be non-linear in nature and it can be at any degree of complexity. Neural networks are composed of simple neuron-like operating elements (neurons) and weighted connections between these elements. The network function is determined largely by the connections between neurons. A neural network can be trained to perform a particular job by adjusting the values of the connections (weights) between neurons. The algorithmic approaches for developing an artificial neural network model for a specific problem exist in literature (Fu 1994; Mathworks 1998).

15.1.3
The Use of Artificial Neural Networks in Environmental Modelling

There are numerous examples of the use of neural networks in environmental modelling. Moreau et al. (1999) embedded neural networks in Lotka-Volterra predator-prey models. Brion and Lingireddy (1997) used neural networks in identification of the sources of microbial contamination. Zhang and Stanley (1997) adopted neural networks for water demand forecasting. Robertson and Morison (1999) attempted to estimate the age of fish automatically with a neural network algorithm that proved successful at least for some fish species. The use of neural network algorithms in modelling and analysis of eutrophication in lakes is also quite promising because of the complex nature of the problem.

Several studies have been carried out on the use of neural networks in eutrophication modelling and Lake Management in recent years. Scardi (1996) used neural networks as to estimate phytoplankton production, Recknagel et al. (1997), Recknagel (1997) and Yabunaka et al. (1997) predicted chlorophyll-a concentration and algal species abundance as a function of sampled water quality parameters. Some zooplankton, such as Rotifers and Diaphanosoma sp., were added as variables to simulate the predator grazing in these studies. Karul et al. (1998a) developed an input-output model where measured water quality parameters were used to estimate chlorophyll-a concentrations. Keiner and Brown (1998) used neural networks to estimate chlorophyll-a concentrations at the ocean surface as an alternative to linear regression methods. Results of all of the above-cited works achieved satisfactory levels of precision. However, there is insufficient information to compare the effectiveness of neural network approaches against the use of multiple regression methods. One notable exception (Karul et al. 1999b) was based on only a single water body.

The main objective of this study was to develop neural network models for different water bodies to simulate eutrophication process. It was thought that the neural network based models that are adequately trained with several environmental factors could be a better approach with more precise predictions than multiple regression methods. A further objective of this study was to compare the performance of the neural network models with that of multiple linear regression models.

15.2
Data and Lakes

Data collected from three very different Turkish water bodies, the Keban Dam Reservoir (KDR), Mogan and Eymir Lakes, have been utilized in this study. Table 15.1 summarizes the basic properties of these water bodies in a comparative way.

Table 15.1. Properties of Freshwater Lakes

Properties	Lake Eymir	Lake Mogan	Keban Dam Reservoir
Lake formation	Formed by the deposition of alluvial material carried by side tributaries.	Formed by the deposition of alluvial material carried by side tributaries.	Artificial reservoir primarily for energy production.
Shape	A riverine pattern characterized by riverine pattern by long and narrow morphology.	Resembles enlarged riverbeds.	Irregular plan view.
Depth	Shallow	Shallow	Deep
Watershed area	970 km^2	970 km^2	64 100 km^2
Average width	300m	1350m	
Center-line length	4.5 km	6 km	151 km
Average water elevation (From sea level)	968.5m	972m	845 m
Average surface area	1.22 km^2	5.43 km^2	191 km^2 at max. water level
Average volume	3.5 million m^3	11.63 million m^3	$30.6*10^9$ m^3
Average depth	3m	2.20 m	21.7 m at max. water elevation
Trophic status	Eutrophic-Dominated by suspended algae	Eutrophic-Dominated by macrophyte	Oligotrophic to eutrophic (No macrophytes)
Secchi depths	0.25-0.70 m	0.2-3.75 m	0.22-5.64 m
Cholorophyll-a	9.02-87.2 0 µg/lt.	0.0-23.8 µg/lt.	1-33.36 µg/lt.
P	0.05-0.57 mg/l as ortho-P	0.021-0.81mg/l as Total-P	0.001-0.081 mg/l as Phosphate

KDR is located in Eastern Anatolian Part of Turkey between northern latitudes of 35°20′ and 38°37′, and eastern longitudes of 38°15′ and 39°52′. Due to highly varying

seasonal hydrological inputs as well as power generation, the reservoir is subject to significant water level fluctuations exposing sediments from the inundated area at various times. Further, mass loads of pollutants entering the reservoir, for example, nitrogen and phosphorus, are highly seasonal. The data show a persistent seasonal metalimnetic oxygen minima, and high spatial heterogeneity with respect to Secchi depth, nutrient concentration levels, turbidity and chlorophyll-a (Soyupak et al. 1998). The data utilized in this study covers the years 1991-1993 (see data of Yemişen et al. (1994) and compilation by Karul (1998b) and (1999a)).

Mogan and Eymir lakes are small and shallow lakes located near Ankara in Central Anatolia. The total watershed of the lakes is about 970 km^2 at the outlet of Eymir. Both of lakes are formed by the inundation of the main canal through deposition of alluvial material. Hence, the lakes are analogous to wide riverbeds. The database that was utilized for Mogan and Eymir Lakes during this study covers the years 1993-1995 (Altınbilek et al. 1995) and was compiled by Karul (1999a). Mogan Lake is situated between the northern latitudes of 39° 28' and 39° 53' and eastern longitudes of 32°30' and 33°00'. Mogan Lake is a shallow eutrophic lake with seasonally dense growth of macrophytes and wide seasonal and diurnal variations in dissolved oxygen concentrations. The main inflow enters the lake through a swamp area and the outflow gives rise to a canal, which enters Eymir Lake. Hydrologically, Lake Eymir is fed by the outflow from Lake Mogan as well as Kışlakçı Creek, a small tributary confluence near the lake outlet. On the bases of OECD Lake classification criteria (Vollenweider and Kerekes 1981), Lake Eymir is eutrophic lake.

126, 31 and 34 valid data sets were obtained from Keban Dam Reservoir (Yemişen et al. 1994), Mogan Lake and Eymir Lake (Altınbilek et al., 1995), respectively. Each data set for KDR included PO_4, NO_3, Alkalinity, Suspended Solids, pH, Temperature, Electrical conductivity, Dissolved Oxygen, Secchi depth, densities of Daphnia species only, and bulk densities of species belonging to Cladocera and Copepoda as input and Chlorophyll-a as output. Data sets for Mongan and Emir Lakes include Total Phosphorus, NO_3, NH_3, Suspended Solids, Temperature, Electrical conductivity, pH, Turbidity, Secchi depth as input and Chlorophyll-a as output.

15.3 Methodology

Data sets available for each water body were collected over several months from the upper 1 m (euphotic zone) of pre-selected stations for each lake. Since the available data was not appropriate to form times series a steady-state approach was adopted. Temporal component was eliminated, since the prime goal of the study was to develop neural network tools for chlorophyll-a concentrations utilizing the data related to governing environmental parameters. Eliminating temporal component was something new when compared to almost the earlier related works (Recknagel et al. 1997; Yabunaka et al. 1997; Karul et al. 1998a).

15.3.1
Artificial Neural Network Approach

An algorithmic approach for developing artificial neural network models for estimating chlorophyll-a concentration in lakes and reservoirs have been discussed in detail previously by Karul et al. (1999b; 2000). A summary of previously established approach is given below.

15.3.1.1
Training Method

A three layer feed-forward neural network model was used. Fig. 15.1 shows the adopted neural network topology for the estimation of output parameters for Keban Dam Reservoir as an example. A tangent-sigmoid transfer function was selected between the input layer and the hidden layer, and a linear transfer function was selected between the hidden layer and the output layer. The Neural Network Toolbox of MatLab by Mathworks Co. (Demuth and Beale 1998) was used during the study.

There are many variations of the backpropogation algorithm and the simplest implementation of it updates the network weight and bias values in the direction in which the performance function decreases most rapidly, i.e. the negative of the gradient. One iteration of the backpropagation algorithm is given by Equation (15.1).

$$x_{k+1} = x_k - \alpha_k g_k \tag{15.1}$$

where x_k is the vector of weights and biases at the k^{th} iteration; α_k is the learning rate at the k^{th} iteration ; g_k is the gradient at the k^{th} iteration.
The Levenberg–Marquardt variation of the backpropagation algorithm was employed in calculation of all neural network weights. The Levenberg–Marquardt algorithm converges faster than other back propagation algorithms and is probably best when there are neurons up to a few hundred. Hagan and Menhaj (1994) give detailed information on the utilization of the Levenberg-Marquardt algorithm and a summary for its implementation is included in Demuth and Beale (1998). The algorithm is:

$$x_{k+1} = x_k - [J^T J + \mu I]^{-1} J^T e \tag{15.2}$$

where x_k is the vector of weights and biases at the k^{th} iteration; J is the Jacobian matrix, which contains first derivatives of the network errors with respect to weights and biases; e is the vector of network errors; I is the identity matrix and μ is a scalar.

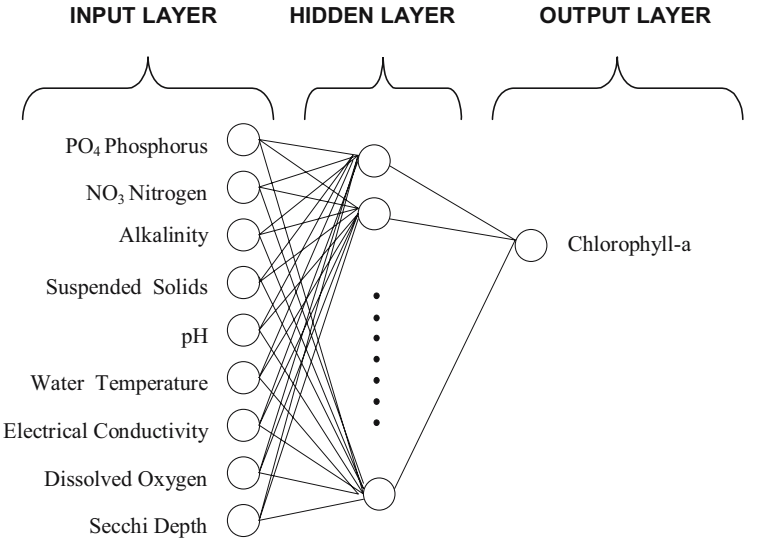


Fig. 15.1. An example of the neural network structure (e.g. KDR) for the estimation of output parameters in case studies

Once trained, the weights and biases of the neural network can be used to generate the output vector a as a function of input vector p, as given in Equation (15.3).

$$a = f2(HW.\ f1(IW.p+b1)+b2) \tag{15.3}$$

where b1 is the bias vector between the input layer and the hidden layer; b2 is the bias vector between the hidden layer and the output layer; IW is the weight matrix between the input layer and the hidden layer; HW is the weight matrix between the hidden layer and the output layer; f1 is the transfer function between the input layer and the hidden layer; f2 is the transfer function between the hidden layer and the output layer; p is the input vector and a is the simulated output vector.

The hyperbolic tangent sigmoid function and linear function are given by equations (15.4) and (15.5) respectively.

$$f1(x) = \frac{2}{1+e^{-2x}} - 1 \tag{15.4}$$

$$f(x) = x \tag{15.5}$$

μ is decreased after each successful step and is increased when an individual step increases the performance function. By this manner, the performance function will always be reduced at each iteration of the algorithm. An initial μ value of 0.001 was used.

15.3.1.2 Data Pre-Processing

To increase the efficiency of training, the network inputs and targets were scaled by normalizing the mean and S.D. of the training set. This process normalizes the input and target values so that they have zero mean and unity S.D. When training is completed, the simulation results are de-normalized by reversing the action.

15.3.1.3 Improving Generalization

A three-layer feed-forward backpropagation neural network with sufficient number of neurons can approximate any function. Thus, one should be aware of the danger that neural network may be memorizing the available data rather than generalizing it, so called over-fitting the data. An over-fitted neural network model typically imitates the data in the training set very successfully but generates a bad estimation for the data not included in the training. For a good generalization, over-fitting should be prevented taking the appropriate measures. Over-fitting can be prevented by utilizing either of the two methods: i) Regularization, and ii) Early stopping. The second method, early stopping, is used in this study to prevent overtraining.

To decide when to stop training, the data set is randomly divided into three sub-sets, one half is used for training, one quarter for validation and the last quarter for testing. The error term, i.e. the difference between measured target values and the calculated values was calculated for the training set, validation set and the test set separately. The error on the validation set will normally decrease during the initial part of the training. However, when the network begins to overfit the data, the validation set error will start to rise. When this increase continues for a predefined number of iterations the training is stopped and the weight values are kept constant. The set is used to compare with the validation set to see if they exhibit a similar behaviour. If the validation errors and test errors do not show a similar behaviour, this may indicate a poor division of data.

Mean square error is the typical performance function used in feed-forward neural networks:

$$MSE = \sum (e_i)^2 / N = \sum (t_i - a_i)^2 / N \tag{15.6}$$

where MSE is the mean square error; N is the number of elements; i is the index for elements; e_i is the error of the i^{th} element; t_i is the target value (measured) for i^{th} element and a_i is the calculated value for i^{th} element.

15.3.2
Multiple Regression Modelling Approach

For each neural network model, a corresponding multiple linear regression model was developed. The aim of using exactly the same data values was to compare the performances of two methods (neural network and multiple linear regression) under exactly the same conditions. To achieve this, the available data was divided into two equal batches, one of which is used to train the neural network while the entire data is used to calculate the regression coefficients. Exactly the same data batch was used to calculate the multiple linear regression models and the regression coefficients were calculated using the entire data. MatLab of Mathworks Co. (Demuth and Beale 1998) was used for all calculations.

15.4
Results

The Figures 15.2, 15.3 and 15.4 give the measured data against predictions utilizing the Neural Network Model for KDR, Eymir and Mogan Lakes respectively. Similarly Figures 15.5, 15.6 and 15.7 presents the measured data against predictions utilizing the Multiple Regression Models for these water bodies. Same data set of each water body was used for both methods.

The multiple regression models are given below:

For KDR:

$$chl\_a_{keban} = 10^{-3.2506.alk + 0.0067.EC - 0.0016.DO + 0.983NO3 - 0.0485.pH - 0.1331.PO4 + 5.3673.Secchi - 0.0323.SS + 0.007.Temp + 0.348} \quad (15.7)$$

For Mogan lake:

$$chl\_a_{mogan} = 10^{-9.57.NH3 + 1.01.NO3 + 0.2831.TP - 0.6682.E.C. + 0.0088.pH - 0.265.Secchi - 0.3215.SS + 0.0248.Temp - 0.4849.Turb - 0.0919} \quad (15.8)$$

For Eymir Lake:

$$chl\_a_{eymir} = 10^{-0.3455.NH3 + 0.4414.NO3 - 0.0807.TP + 0.1388.E.C. + 0.0014.pH - 0.1691.Secchi - 2.667.SS + 0.0009.Temp - 0.0737.Turb - 0.0025} \quad (15.9)$$

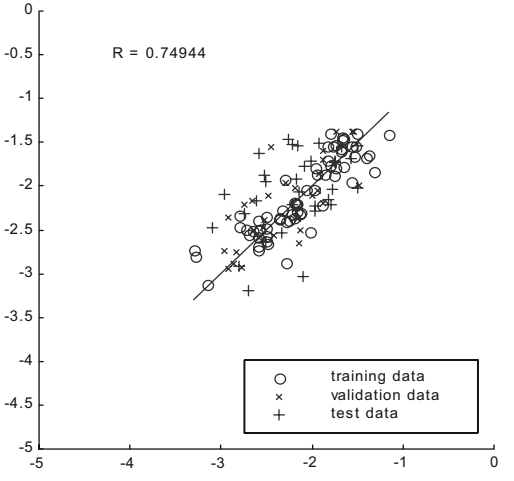


Fig. 15.2. The regression plot for the KDR neural network model results showing training, validation and test data

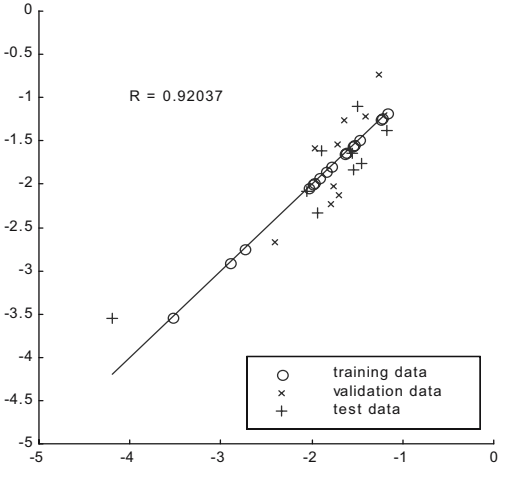


Fig. 15.3. The regression plot for Mogan Lake neural network model results

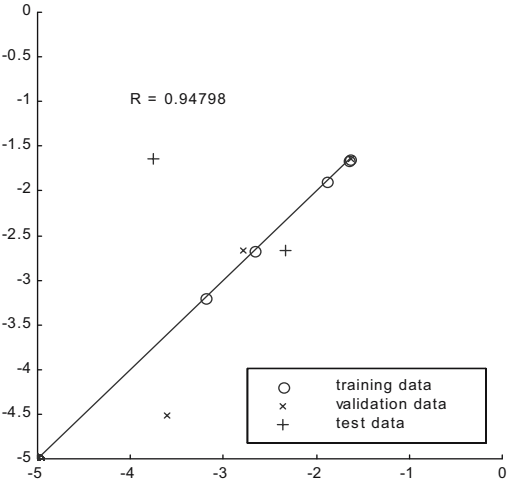


Fig. 15.4. The regression plot for Eymir Lake neural network model results

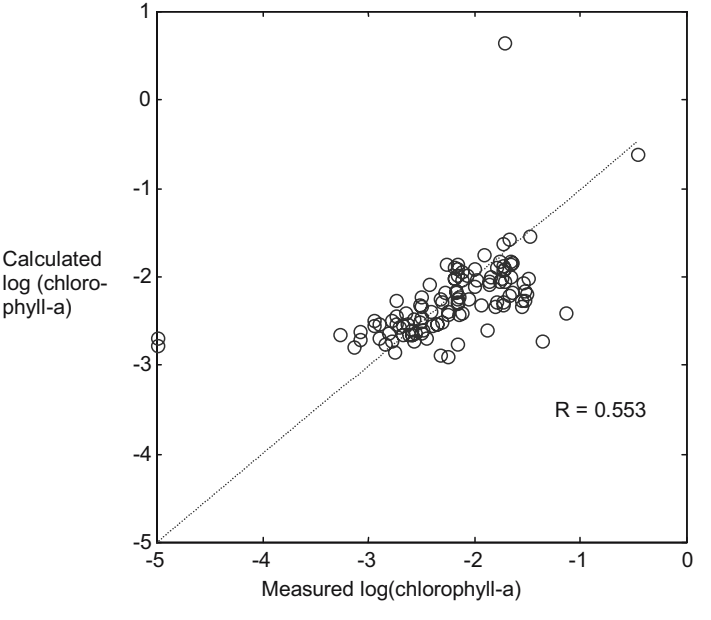


Fig. 15.5. Multiple regression results for the KDR

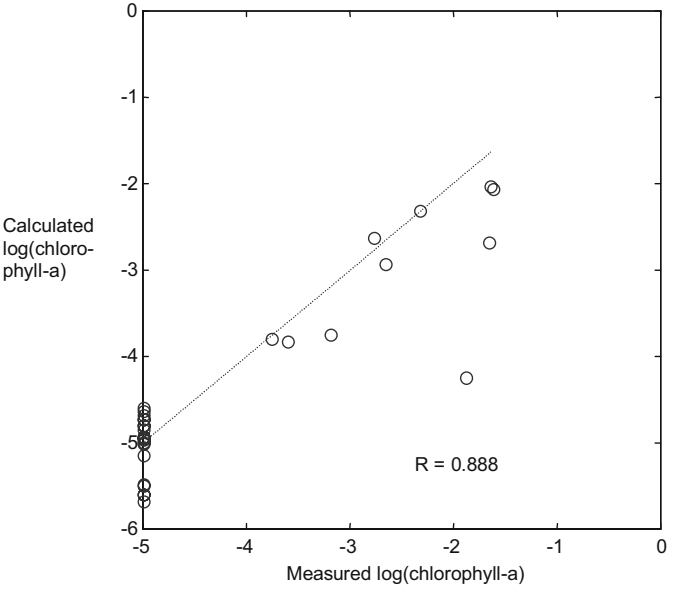


Fig. 15.6. Multiple regression results for Mogan Lake

where chl-a is the chlorophyll-a concentration, mg/l; alk is the alkalinity, mg/l as $CaCO_3$; EC is the electrical conductivity,μmho; DO is the dissolved oxygen concentration, mg/l; NO_3 is the nitrate concentration, mg/l; PO_4 is the phosphate concentration, mg/l; Secchi is the Secchi depth, m ; SS is the suspended solids concentration ,mg/l and Temp is the water temperature, degree Celcius, NH_3 is the ammonia concentration, mg/l and Turb is the turbidity, NTU.

15.5 Conclusions and Recommendations

15.5.1 Conclusions

Performances of artificial neural network models:
Assessment of the performances of the developed artificial neural network models was made possible with the help of a group of regression plots (Figures 15.2, 15.3 and 15.4). The linear regression coefficients were 0.75, 0.95 and 0.92 for Keban Dam Reservoir, Mogan Lake and Eymir Lake respectively. The better results for Mogan and Eymir Lakes as compared to KDR were attributed to their relatively

much smaller size and homogenous characteristics. However, an R-value such as 0.75 for a very large water body with high temporal and spatial variability can still be assumed as reasonably acceptable.

Performances of multiple regression models:
Similar to artificial neural network models, assessment of the performances of the developed multiple regression models were made possible with the help of a group of regression plots (Figures 15.5, 15.6 and 15.7). The linear regression coefficients were 0.55, 0.88 and 0.69 for Keban Dam Reservoir, Mogan Lake and Eymir Lake respectively. Again, R-values of regression plots were relatively higher for Mogan and Eymir Lakes as compared to R-value for KDR. However, R-value of multiple regression plot was significantly lower than R-value of artificial neural network plot for each water body.

Comparisons of the Performances of Artificial Neural Networks and multiple regression models:
Examination and comparison of the figures (with the calculated R-values) related to the regression plots and for neural network model results gives an obvious impression on the superiority of neural network model for KDR, Mogan Lake and Eymir Lake. So it was concluded that the neural network model seemed to predict chlorophyll-a with a better performance than that of the selected multiple regression models for the examined water bodies of entirely different character.

As a concluding remark it can be stated that, there is a potential in artificial neural network approach to be used as a modelling tool to estimate major parameters of eutrophication (i.e. chlorophyll-a) because of its inherent property of being able to be trained for very complex and non-linear systems. Neural network can be trained to recognize the environment to predict the system's response to the conditions of the environment even in highly variable water bodies with respect to location and time.

15.5.2 Recommendations

Since this study had the main intentions of getting initial information related to i) the performances of artificial neural networks in estimating chlorophyll-a concentrations in different water bodies and ii) comparing the performances of artificial neural networks with that of simple regression methods, sophisticated statistical evaluation methods were not planned to be employed as comparison tools at the beginning. However, it is recommended to apply such tools in future to strengthen the conclusions derived from this particular study. It is further suggested that the performance of artificial neural networks should be tested utilizing the relevant data obtained from several other water bodies.

Acknowledgments

Middle East Technical University through research fund AFP-97-03-11-04 supported this study. Data used in the studies were provided by State Hydraulic Works of Turkey and Middle East Technical University, Department of Environmental Engineering through projects supported by TUBİTAK.

References

Altınbilek D, et al. (1995) Gölbaşı Mogan and Eymir Lakes Water Resources and Environmental Management Planning Project, Final Report, Middle East Technical University, Ankara, Turkey

Bartsch AF, Gakstatter JH (1978) Management Decisions for Lake Systems on a Survey of Trophic Status, Limiting Nutrients, and Nutrient Loadings in American-Soviet Symposium on Use of Mathematical Models to Optimize Water Quality Management, 1975, U.S. EPA Office of Research and Development, Environmental Research laboratory, Gulf Breeze, FL, EPA-600/9-78-024, 372-394

Benndorf J, Recknagel F (1982) Problems of application of the ecological model SALMO to lakes and reservoirs having various trophic states, Ecol. Modelling, 17, 129-145

Brion GM, Lingireddy S (1997) A neural networks approach to identify sources of microbial contamination, CSCE/ASCE Env. Eng. Conf. Proceedings, Edmonton, Alberta, anada, 1321-1332

Canfield DC, Langeland KA, Maceina MJ, Haller WT, Shireman JV, Jones JR (1983) Trophic state classification of lakes with aquatic macrophytes, Can. Jour. Fish Aquatic Science, 40, 1713-18

Canfield DC, Shireman JV, Coole DE, Haller WT, Watkins CE, Maceina MJ (1984) Prediction of chlorophyll a concentrations in Florida Lakes: Importance of aqautic macrophytes, Can. Jour. Fish Aquatic Science, 41, 409-501

Demuth H, Beale M (1998) Neural Network Toolbox User's Guide, The MathWorks Inc., Natick, MA

Dillon PJ, Rigler FH (1974) The phosphorus-chlorophyll relationship in lakes, Limnol. Oceanogr., 19, 767-773

Fu LM (1994) Neural Networks for Computer Intelligence, McGraw-Hill, Inc

Hagan MT, Menhaj M (1994) Training feedforward networks with the Marquardt algorithm, IEEE Trans.on Neural Networks, 5, 989-993

Jorgensen SE (1976) A eutrophication model for a lake, Ecol. Model., 2, 147-162

Karul C, Soyupak S, Germen E (1998a) A new approach to mathematical water quality modelling in Reservoirs: Neural Networks, Int. Rev. of Hydrobiol., 83, 689-696

Karul C, Soyupak S, Güven E, Aydoğan A, Alp E (1998b) Limnolojik veri tabanı geliştirilmesi ve TCP/IP üzerinden WWW arayüzü ile işletilmesi: Keban Baraj Gölü Veri Tabanı Örneği, DSİ Çevre Semineri, Fethiye,Turkey (in Turkish)

Karul C (1999a) Development of An Artificial Neural Network Model for the Estimation of Chlorophyll-a in Lakes, PhD Thesis, METU, Department of Environmental Engineering, Ankara, Turkey

Karul C, Soyupak S, Yurteri C (1999b) Neural network models as a management tool in lakes, Hydrobiologia , 408-409, 139 - 144

Karul C, Soyupak S, Çilesiz AF, Akbay N, Germen E (2000) Case studies on the use of neural networks in eutrophication modeling, Ecol. Model. ,134 (2-3), 145 – 152

Keiner LE, Brown CW (1998) A neural network as a non-linear chlorophyll estimation algorithm", http://orbit19i/nesdis/noaa.gov/~lkeiner/seanam_neural/seabam2.htm

Moreau Y, Louies S, Vandewalle J, Brenig L (1999) Embedding recurrent neural networks into predator-prey models, Neural Networks, 12, 237-245.

Rast W, Lee GF (1978) Summary Analysis of the North American Project (US Portion) OECD Eutrophication Project: Nutrient Loading-Lake Response Relationships and Trophic State Indices, USEPA Corvallis Environmental Research Laboratory , Corvallis, OR,EPA-600/3-78-008.

Recknagel F, Petzoldt T, Jacke O, Krusche F (1994) Hybrid expert system DELAQUA: a toolkit for water quality control of lakes and reservoirs, Ecol. Model., 71, 17-36.

Recknagel F, French M, Harkonen P, Yabunaka K (1997) Artificial neural network approach for modelling and prediction of algal blooms, Ecol. Model., 96, 11-28.

Recknagel F (1997) ANNA - Artificial Neural Network model predicting species abundance and succession of blue-green Algae. Hydrobiologia 349, 47-57.

Robertson SG, Morison AK (1999) A trial of artificial neural networks for automatically estimating the age of fish, Mar. Freshwater Res., 50, 73-82.

Ryding S, Rast W (1989) The Control of Eutrophication of Lakes and Reservoirs, Parthenon Publishing Co., UNESCO.

Scardi M (1996) Artificial neural networks as empirical models for estimating phytoplankton production, Mar. Ecol. Ser., 139, 289-299.

Smith VH, Shapiro J (1981) A Retrospective Look at the Effects of Phosphorus Removal in Lakes,in Restoration of Lakes and Inland Waters, USEPA, Office of Water Regulations and Standards, Washington, DC, EPA-440/5-81-010.

Soyupak S, Yemişen D, Mukhallalati L, Erdem S, Akbay N, Yerli S (1998) The spatial and temporal variability of limnological properties of a very large and deep reservoir, Journal of International Review of Hydrobiology, 83, 183-190.

Vollenweider RA, Kerekes JJ (1981) Background and Summary Results of the OECD Cooperative Program on Eutrophication , Int. Symp. on Inland Waters and Lake Restoration, U.S. EPA, Washington D.C., 25-36.

Whitehead PG, Hornberger GM (1984) Modelling algal behavior in the River Thames, Wat. Res., 18(8), 945-953.

Yabunaka K, Hosomi M, Murakami A (1997) Novel application of a back-propagation artificial neural network model formulated to predict algal bloom, Water science and Technology., 36, 89-97.

Yemişen D (1994) Keban Baraj Gölü ve Havzası Çevre Sorunları Projesi - Final Raporu - Keban Baraj Gölü Sularının Fiziksel, Kimyasal ve Biyolojik Özellikleri-Keban'da Ötrofikasyon-Hidrodinamik Modelleme Sonuçları ve Su Kalitesi Modelleme Sonuçları-Çözüm Önerileri, TÜBİTAK DEBAG 124/G Projesi, ODTÜ-DSİ Genel Müdürlüğü ve DSİ 9. Bölge, Elazığ, Turkey(in Turkish).

Zhang Q, Stanley SJ (1997) The artificial neural network modelling approach for water demand forecasting in Edmonton, CSCE/ASCE Env. Eng. Conf. Proceedings, Edmonton, Alberta, Canada, 1333-1344.

Chapter 16

Artificial Neural Network Approach to Unravel and Forecast Algal Population Dynamics of Two Lakes Different in Morphometry and Eutrophication

F. Recknagel · A. Welk · B. Kim · N. Takamura

16.1 Introduction

Limnological time-series of the unstratified shallow, eutrophic Lake Kasumigaura (Japan) and the stratified deep, mesotrophic Lake Soyang (Korea) were used for a comparative study of phytoplankton population dynamics by super- and non-supervised artificial neural networks (ANN).

Water quality data of Lake Kasumigaura from 1984 to 1993 revealed lasting hypertrophic conditions over the study period with mean total phosphorus concentrations of 118 µg/l and recurrent blue-green algal blooms in summer (Takamura *et al.* 1992). By contrast water quality data of Lake Soyang from 1988 to 2000 indicated a mean total phosphorus concentration of 16 µg/l with a temporary shift from mesotrophic to eutrophic conditions in the late 1980s and a consolidated mesotrophic state since the late 1990s in response to varying intensities of fish farming (Kim 2002). The change from mesotrophic to eutrophic conditions was accompanied with the changing dominance from dinoflagellates to blue-green algae during late summer blooms which were triggered by high nutrient loadings during the monsoon season (Heo and Kim 1997; Kim et al. 2000; Kim 2002). Both lakes experience similar temperate climate with monsoonal rain in mid-summer. Using real lake data the present research aimed at: (1) forecasting seasonal succession of blue green algae and diatom populations in both lakes and determination of the populations' sensitivities against physical and chemical lake properties by means of recurrent supervised ANN; (2) analysing complex interactions between algal populations and seasons, pH and nutrient conditions, underwater light and temperature conditions by means of non-supervised ANN. Results from the sensitivity analysis by supervised ANN were brought into a context with data ordination and clustering by non-supervised ANN in order to test hypotheses on complex interactions of algal populations with environmental conditions as postulated by Reynolds (1984).

Ecological time-series data of lakes have previously been ordinated and clustered by conventional multivariate statistics (e.g. Varis et al. 1989; Varis 1991; Van Tongeren et al. 1992) but failed to cope with multiple non-linear nature of data. By contrast data ordination and clustering by non-supervised ANN (Kohonen 1989; Kohonen 1995) proves to be applicable to highly complex and non-linear data including limnological time-series (e.g. Chon et al. 1996; Recknagel et al. 2004).

The results show that recurrent supervised ANN allow 7-days-ahead forecasts of seasonal succession and abundances of blue-green algae and diatoms in quite distinctive lakes with reasonable accuracy. The sensitivity curves from supervised ANN complemented well the ordination and clustering of the two algal populations regarding their temperature, nitrogen, phosphorus preferences as well as pH tolerances by means of non-supervised ANN. These results revealed from data corresponded well with related hypotheses postulated by Reynolds (1984). The pattern analysis of periods with distinctively different water quality conditions of the two lakes by non-supervised ANN has discovered behaviours of phyto- and zooplankton likely in response to according management efforts.

16.2 Materials and Methods

16.2.1 Study Sites and Data

Lake Kasumigaura is situated in the southeastern part of Japan and receives flow from 56 rivers and streams. Its catchment area of 2135 km^2 consists of paddy areas but is largely urbanised and industrialised. The lake was turned from a brackish into a freshwater lake 5 years after a floodgate to the Pacific Ocean was implemented in 1963.

Tab. 16.1. General characteristics of Lake Kasumigaura and Lake Soyang

	Lake Kasumigaura	Lake Soyang
Surface area km^2	219.9	45
Maximum volume km^3	662	2900
Maximum depth m	7	110
Mean depth m	3.9	42
Water residence time years	0.55	0.7
Catchment area km^2	1597	2675
Circulation Type	non-stratified	warm monomictic

Lake Soyang is situated in the northeastern part of South Korea and fed by the Soyang River contributing 90% of the inflowing water. Nutrient loadings to Lake Soyang are predominantly caused by non-point sources from paddy and forest areas, and temporarily by in-lake fish farming using net cages.

Tab. 16.1 summarises characteristics of the two lakes. Tab. 16.2 provides details of the limnological databases of the two lakes.

Data of Lake Kasumigaura were collected with a column sampler of 2m at the centre of the Takahamairi Bay. Data of Lake Soyang were collected in meter steps at the central station and averaged over the upper 10 m for the present study. As the measurement intervals of the raw data from both lakes were highly irregular and sampling dates different for physical, chemical and biological data the data was interpolated to create consistent daily values as required for the development of ANN models.

Tab. 16.2. Limnological properties reflected by the databases of Lake Kasumigaura and Lake Soyang

	Lake Kasumigaura (1984 – 1993)	**Lake Soyang (1988 – 2000)**
Limnological Variables	**Mean / Min / Max**	**Mean / Min / Max**
PO_4 µg/l	14.44 / 1 / 235	3.4 / 0.15 / 20
NO_3 mg/l	0.52 / 0.001/ 2.38	1.02 / 0.3 / 2.2
Si mg/l	3.66 / 0.015 / 12.49	
Chl-*a* µg/l	73.05 / 0.69 / 279.5	3.7 / 0.4 / 45.1
DO mg/l	10.23 / 4.88 / 18.21	9.36 / 5.3 / 13.1
Turb. NTU		1.36 / 0.5 / 10.35
Secchi Depth m	0.87 / 0.28 / 3.8	4.12 / 0.7 / 10
pH	8.75 / 7.12 / 10.13	7.3 / 6.2 / 9.1
Water Temperature °C	16.48 / 2.1 / 32	15.2 / 4.6 / 29.5
Phytoplankton cells/ml:		
Anabaena	6008 / 1 / 112112	465 / 1 / 17000
Oscillatoria	20160 / 1 / 502302	
Microcystis	38563 / 1 / 644117	162 / 1 / 9130
Cyclotella	5160 / 1 / 75420	
Asterionella		670 / 1 / 20600
Zooplankton ind./l:		
Cladocera	170 / 1/ 2446	4 / 1 / 56
Copepoda	156 / 1 / 640	7 / 1 / 80
Rotifera	229/ 1 /2542	32 / 1 / 47

16.2.2 Methods

A recurrent supervised ANN (Fig. 16.1) was applied to predict 7-days-ahead seasonal succession of *Microcystis* and *Cyclotella* for Lake Kasumigaura, and of *Anabaena* and *Asterionella* for Lake Soyang. Recurrent supervised ANN were introduced by Pineda (1987) mimicking the principles of deterministic modelling by ordinary differential equations. They consider both, current external inputs as well as feedback inputs of copied neuron weights at time *t*-1 in order to determine

current weights of neurons at time *t*. They prove to be very efficient for time series modelling (e.g. Walter *et al.* 2001; Jeong, Recknagel and Joo 2003).

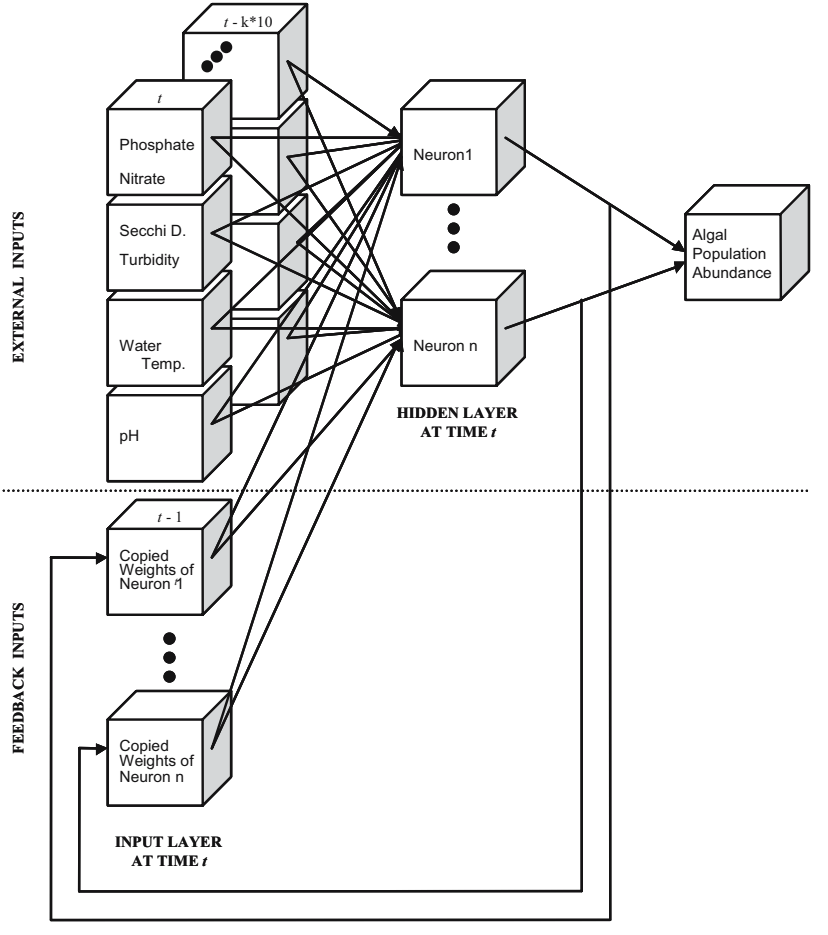


Fig. 16.1. Structure of the recurrent supervised ANN

For Lake Kasumigaura the ANN were trained with daily input values for PO_4, NO_3, Si, Secchi depth, pH and water temperature and daily output values for *Microcystis* and *Cyclotella* of the years 1984 to 1985 and 1987 to 1993. The prediction results for 1986 were validated with independent daily input and output values not used for ANN training, and assessed by the mean square errors (MSE). A comprehensive sensitivity analysis was conducted by means of the recurrent supervised ANN to discover relationships between the input variables and the *Microcystis* and *Cyclotella* populations.

For Lake Soyang we trained the ANN with daily input values for PO_4, NO_3, Secchi depth, turbidity, pH and water temperature and daily output values for

Anabaena and *Asterionella* of the years 1990 to 1992 and 1994 to 2000. The prediction results for 1997 were validated with independent daily input and output values not used for ANN training, and assessed by the r^2 values of the linear

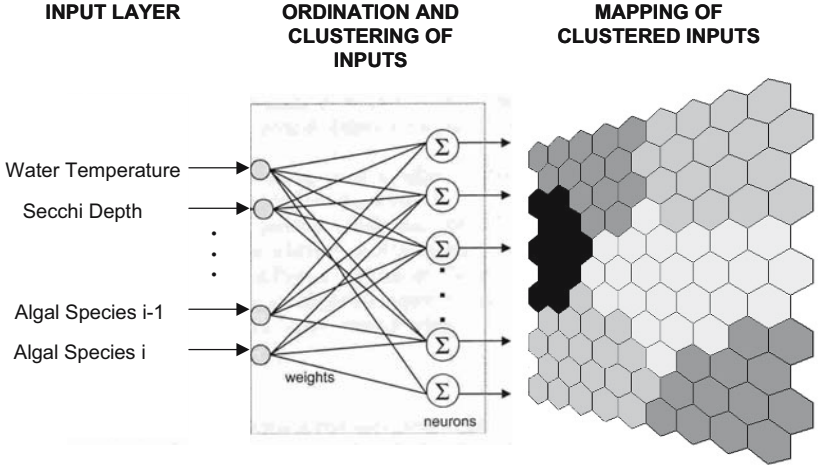
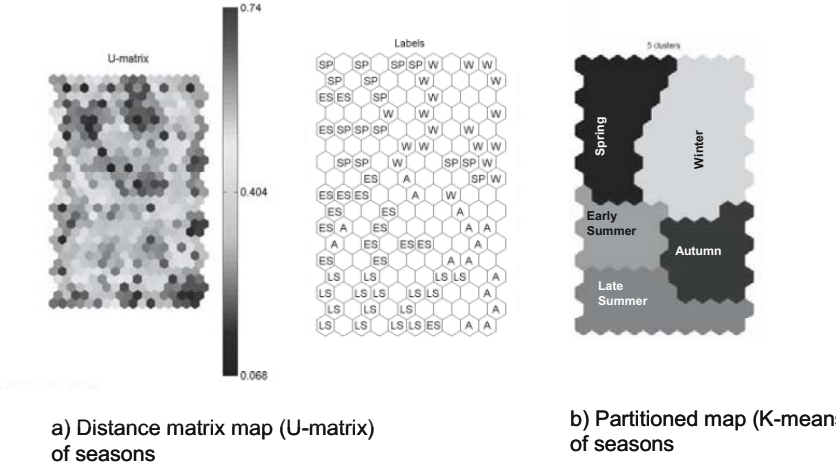


Fig. 16.2. Structure of the non-supervised ANN



a) Distance matrix map (U-matrix) of seasons

b) Partitioned map (K-means) of seasons

Fig. 16.3. Ordination and clustering of seasons of the year of Lake Kasumigaura by means of non-supervised ANN represented as distance matrix map (a) and partitioned map (b)

equation without intercept. A comprehensive sensitivity analysis was conducted by means of the recurrent supervised ANN to discover relationships between the input variables and the *Anabaena* and *Asterionella* populations.

A non-supervised ANN as introduced by Kohonen (1989) was applied according to Kohonen (1995) to ordinate, cluster and map water quality and

phytoplankton data of both lakes with respect to seasons and ranges of nutrients, pH and water temperature conditions (see Fig. 16.2).

As a result of the training of the non-supervised ANN by means of the normalised input data the Euclidian distance between the inputs is calculated and visualised as a distance matrix (U-matrix) and a partition map (K-means). Whilst dissimilar input patterns map onto different regions of the input space, similar patterns are clustered to groups.

The Fig. 16.3 shows the clustering and mapping of the seasons of the year for Lake Kasumigaura. Tab. 16.3 summarises the criteria for ordination and clustering of time-series data of Lake Kasumigaura and Lake Soyang by means of the non-supervised ANN.

Tab.16.3. Classification criteria for ordination and clustering of time-series data of lake Kasumigaura and Lake Soyang by non-supervised ANN

Classification Criteria	Lake Kasumigaura	Lake Soyang
Seasons:		
Spring	15th March to 30th May	15th March to 30th May
Early Summer	1st June to 30th July	1st June to 30th July
Late Summer	1st August to 30th September	1st August to 30th September
Autumn	1st October to 30th November	1st October to 30th November
Winter	1st December to 14th March	1st December to 14th March
Water Quality Ranges:		
PO_4-P	< 5; 5>= and < 25; > 25	< 5; 5>= and < 25; > 25
NO_3-N	< 0.5; 0.5>= and < 1; >= 1	< 1; 1>= and < 1.5; >= 1.5
pH	<7.5; 7.5<= and < 8.5; >= 8.5	<7; 7<= and < 8.5; >= 8.5
Secchi Depth	<0.75; 0.75<= and < 1.5; >=1.5	<0.75; 0.75<= and < 1.5; >=1.5
Water Temperature	< 15; 15>= and < 20; >= 20	< 15; 15>= and < 20; >= 20

16.3 Results

16.3.1 Forecasting Seasonal Algal Abundances and Succession

The recurrent supervised ANN (Fig. 16.1) were specifically designed and trained for the two lakes in order to forecast abundances of representative blue-green algae and diatom populations for 7-days-ahead. The Fig. 16.4 shows forecasting results for *Microcystis* (Fig. 16.4, top, left) and *Cyclotella* (Fig. 16.4, bottom, left) of the testing year 1986 of Lake Kasumigaura. Whilst the predicted timing and magnitude of the summer peak of *Microcystis* corresponded well with the measured data ($r^2 = 0.9$), the predicted timing of the spring peak of *Cyclotella* was 3 weeks too early with the same magnitude as observed and a $r^2 = 0.4$. The ANN also forecasted slight summer and autumn peaks of *Cyclotella* even though only a similar autumn peak was observed in 1986.

The forecasting results of the summer peak of *Anabaena* (Fig. 16.4, top, right) for Lake Soyang was reasonable regarding both magnitude and timing with a r^2 = 0.6. However the predicted timings of two spring peaks of *Asterionella* (Fig. 16.4, bottom, right) for Lake Soyang were several weeks too early resulting in a r^2 = 0.3.

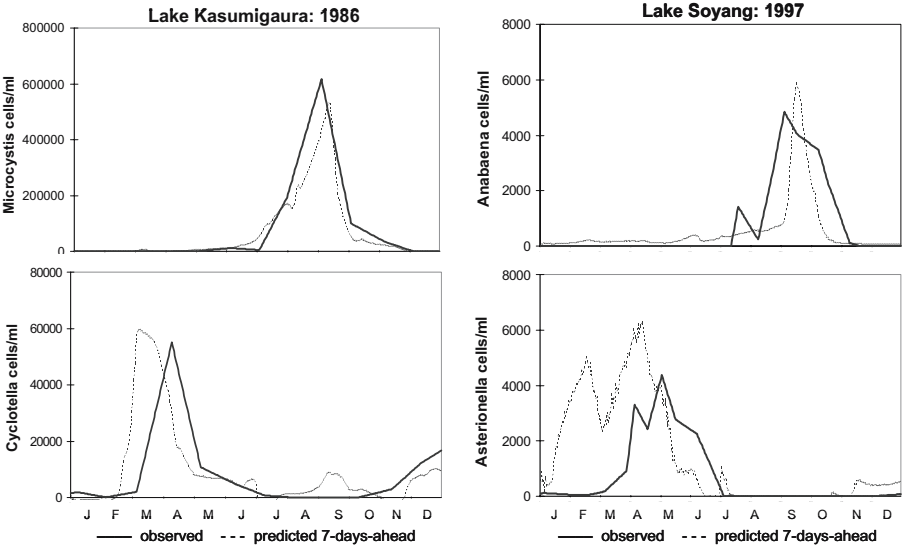


Fig. 16.4. 7-days-ahead forecasting of *Microcystis* and *Cyclotella* for Lake Kasumigaura in 1986 (left column), and of *Anabaena* and *Asterionella* for Lake Soyang in 1997 (right column)

16.3.2 Relationships between Algal Abundances and Water Quality Conditions

Sensitivity analyses by means of the recurrent supervised ANN (Fig. 16.1) as well as ordination and clustering by means of non-supervised ANN (Fig. 16.2) were carried out based on data from 1984 to 1993 of Lake Kasumigaura and data from 1988 to 2000 of Lake Soyang in order to study complex relationships between blue-green algae and diatom populations, and water quality conditions (see Tab. 16.3). The Fig. 16.5 illustrates relationships between water temperature and *Microcystis* and *Cyclotella* in Lake Kasumigaura, and Fig. 16.6 between water temperature and *Anabaena* and *Asterionella* in Lake Soyang. These figures clearly show that in both lakes blue-green algae *Microcystis* and *Anabaena* reach their highest abundances at temperatures higher than 20° C but diatoms *Cyclotella* and *Asterionella* at temperatures below 16° C. In Figs. 16.7 and 16.8 relationships between pH and: (1) *Microcystis* and *Cyclotella* in Lake Kasumigaura, and (2) *Anabaena* and *Asterionella* in Lake Soyang are represented. Even though the pH

ranges (Tab. 16.2) indicate distinctive alkaline conditions of Lake Kasumigaura but neutral to alkaline conditions of lake Soyang,

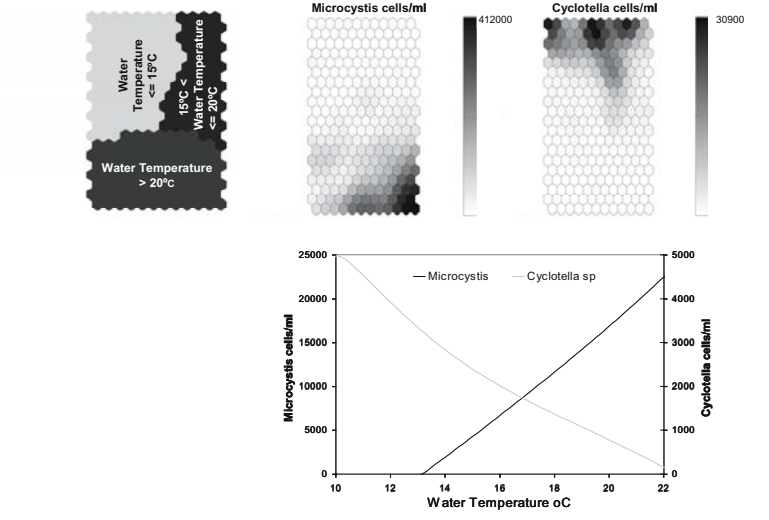


Fig. 16.5. Clustering of *Microcystis* and *Cyclotella* abundances regarding temperature classes in Lake Kasumigaura using non-supervised ANN (top) and sensitivity curves of *Microcystis* and *Cyclotella* abundances over the temperature range of Lake Kasumigaura using supervised ANN (bottom)

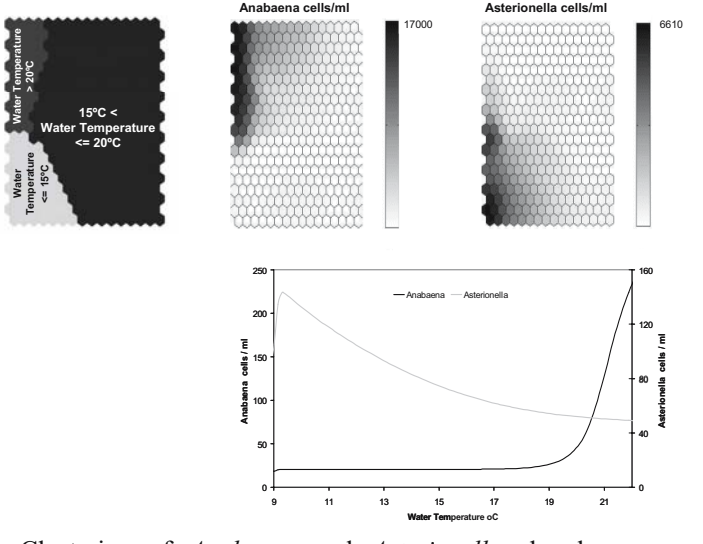


Fig. 16.6. Clustering of *Anabaena* and *Asterionella* abundances regarding temperature classes in Lake Soyang using non-supervised ANN (top) and sensitivity curves of *Anabaena* and *Asterionella* abundances over the temperature range of Lake Soyang using supervised ANN (bottom)

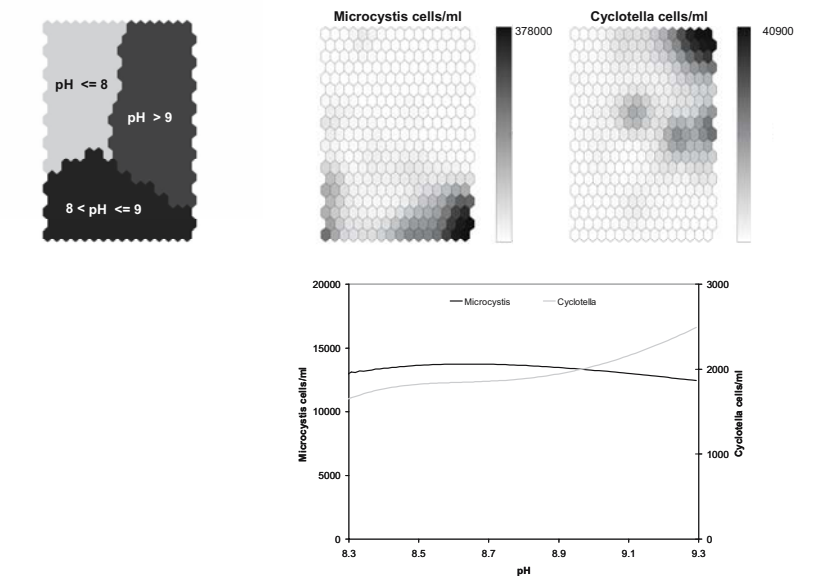


Fig. 16.7. Clustering of *Microcystis* and *Cyclotella* abundances regarding pH classes in Lake Kasumigaura using non-supervised ANN (top) and sensitivity curves of *Microcystis* and *Cyclotella* abundances over the pH range of Lake Kasumigaura using supervised ANN (bottom)

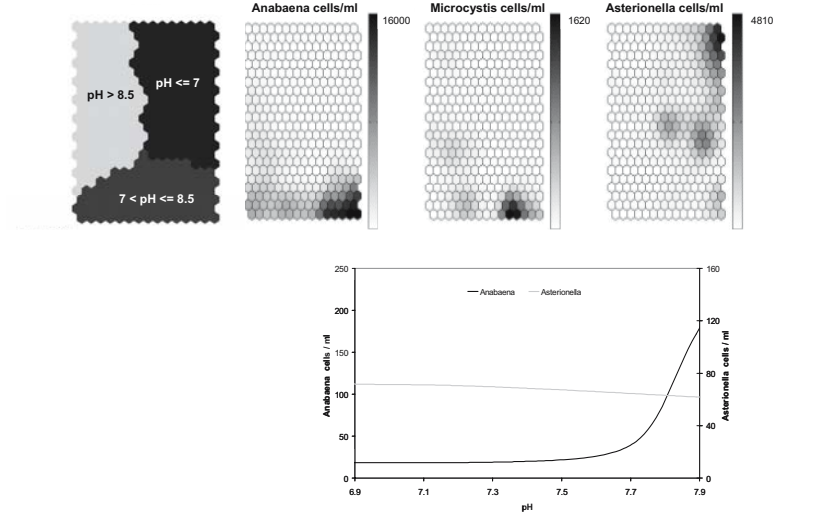


Fig. 16.8. Clustering of *Anabaena* and *Asterionella* abundances regarding pH classes in Lake Soyang using non-supervised ANN (top) and sensitivity curves of *Anabaena* and *Asterionella* abundances over the pH range of Lake Soyang using supervised ANN (bottom)

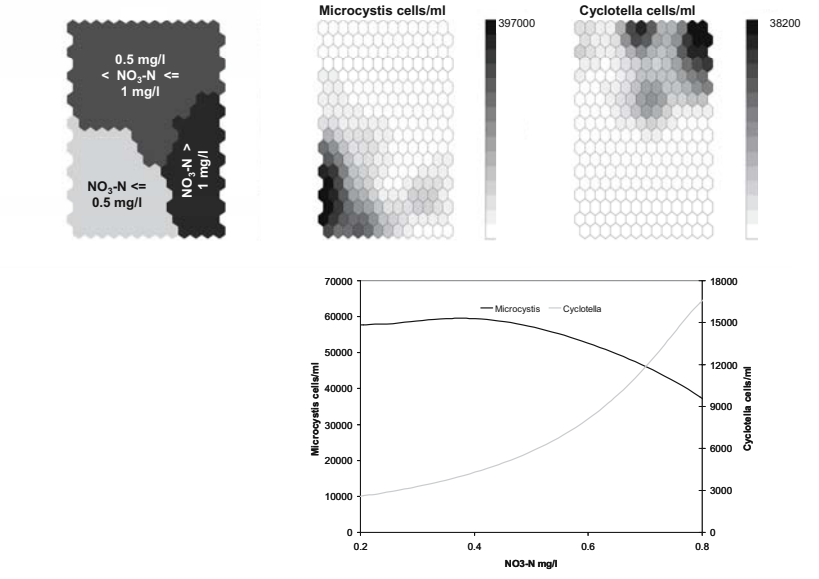


Fig. 16.9. Clustering of *Microcystis* and *Cyclotella* abundances regarding NO_3-N classes in Lake Kasumigaura using non-supervised ANN (top) and sensitivity curves of *Microcystis* and *Cyclotella* abundances over the NO_3-N range of Lake Kasumigaura using supervised ANN (bottom).

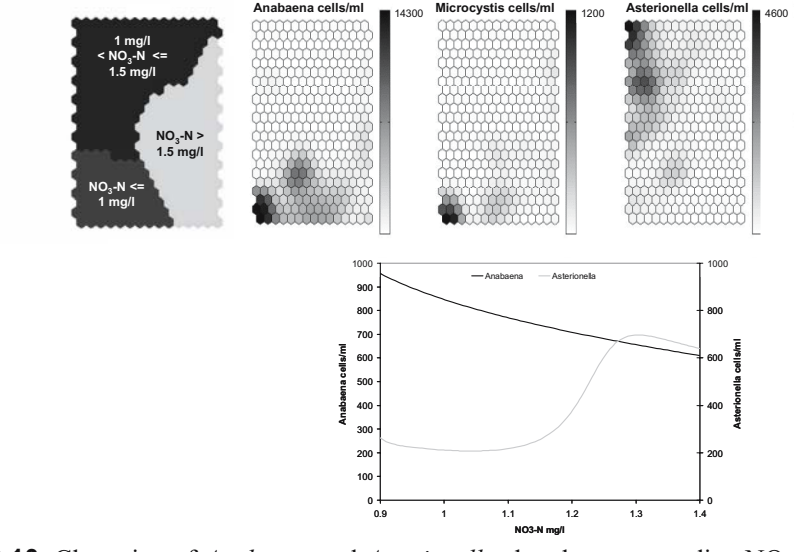


Fig. 16.10. Clustering of *Anabaena* and *Asterionella* abundances regarding NO_3-N classes in Lake Soyang using non-supervised ANN (top) and sensitivity curves of *Anabaena* and *Asterionella* abundances over the NO_3-N range of Lake Soyang using supervised ANN (bottom)

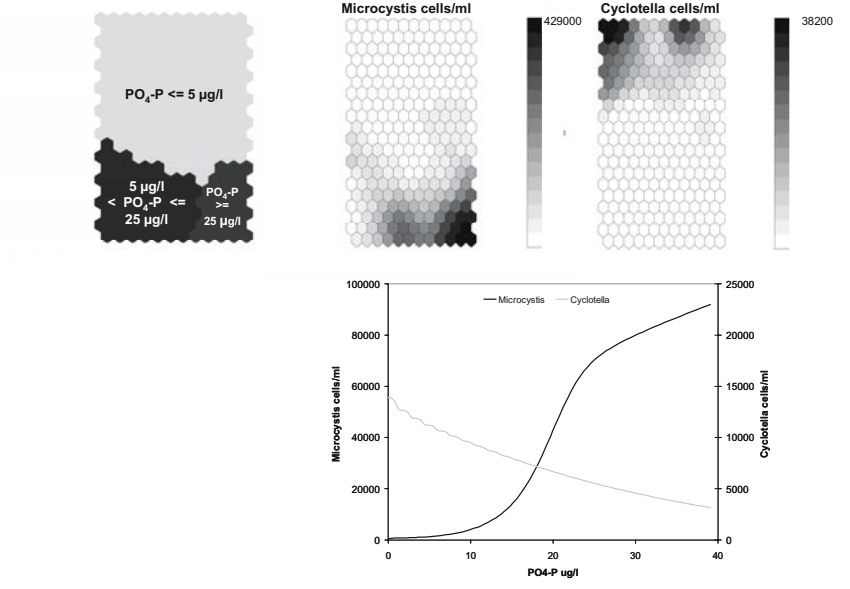


Fig. 16.11. Clustering of *Microcystis* and *Cyclotella* abundances regarding PO_4-P classes in Lake Kasumigaura using non-supervised ANN (top) and sensitivity curves of *Microcystis* and *Cyclotella* abundances over the PO_4-P range of Lake Kasumigaura using supervised ANN (bottom)

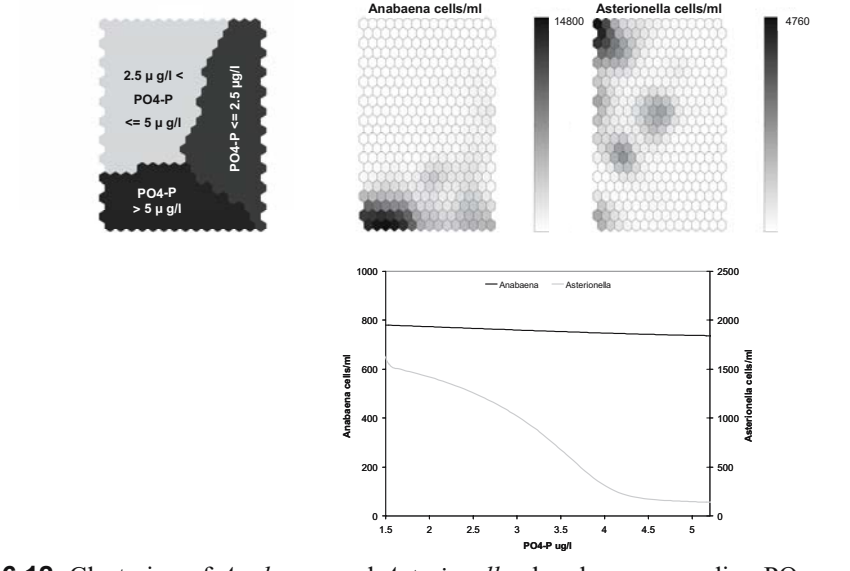


Fig. 16.12. Clustering of *Anabaena* and *Asterionella* abundances regarding PO_4-P classes in Lake Soyang using non-supervised ANN (top) and sensitivity curves of *Anabaena* and *Asterionella* abundances over the PO_4-P range of Lake Soyang using supervised ANN (bottom)

the results in Figs. 16.7 and 16.8 demonstrate the coincidence of high abundances of blue green algae and high pH values of 8 to 9 in Lake Kasumigaura and 7 to 8.5 in Lake Soyang. By contrast the diatom *Cyclotella* in Lake Kasumigaura peaks at pH of 9 to 9.3 whilst the diatom *Asterionella* in Lake Soyang seems to occur at neutral to slightly acidic conditions. The Figs. 16.9 and 16.10 reflect preferences of diatoms to higher concentrations of NO_3-N in both lakes but tolerance of blue-green algae to relatively low concentrations of NO_3-N. The relationships of algal groups to PO_4-P concentrations show the opposite trend compared to NO_3-N for both lakes. The diatoms Cyclotella in Lake Kasumigaura (Fig. 16.11) and Asterionella in Lake Soyang (Fig. 16.12) reach highest abundances at relative low PO_4-P concentrations. However the blue-green algae Microcystis in Lake Kasumigaura (Fig. 16.11) and Anabaena in Lake Soyang (Fig. 16.12) show a distinct preference for highest PO_4-P concentrations.

16.3.3 Relationships between Algal Abundances, Seasons and Water Quality Changes

Ordination and clustering by means of non-supervised ANN (Fig. 16.2) were carried out based on data from 1984 to 1986 and from 1987 to 1989 of Lake Kasumigaura, and data from 1992 to 1993 and 1998 to 1999 of Lake Soyang in order to study complex relationships between functional algal groups, annual seasons (see Tab. 16.3) and water quality changes. The period from 1984 to 1986 of Lake Kasumigaura was selected to reflect conditions of relative PO_4-P sufficiency but NO_3-N deficiency, and 1987 to 1989 because of relative NO_3-N sufficiency but PO_4-P deficiency. The period from 1992 to 1993 of Lake Soyang was selected to reflect conditions of relative PO_4-P sufficiency but NO_3-N deficiency as a result of intensive fish farming, and 1998 to 1999 because of relative NO_3-N sufficiency but PO_4-P deficiency as a result of terminated fish farming.

The Fig. 16.13 shows the seasonal abundance clusters of the green algae *Scenedesmus* and the diatom *Cyclotella* for two periods with distinctive nutrient conditions in Lake Kasumigaura. The results suggest that both algae experience a shift of their predominant occurrence from spring in the period 1984 to 1986 to winter in period 1987 to 1989 with two times higher abundances in the second period. The Fig. 16.14 shows patterns of two blue-green algae where the highest abundance of *Microcystis* shifts from late summer to autumn and declines by 50% between periods 1 and 2. A different trend can be observed in Fig. 16.14 for *Oscillatoria* that shifts its dominance from spring in period 1 to late summer in period 2 by tripling its abundance.

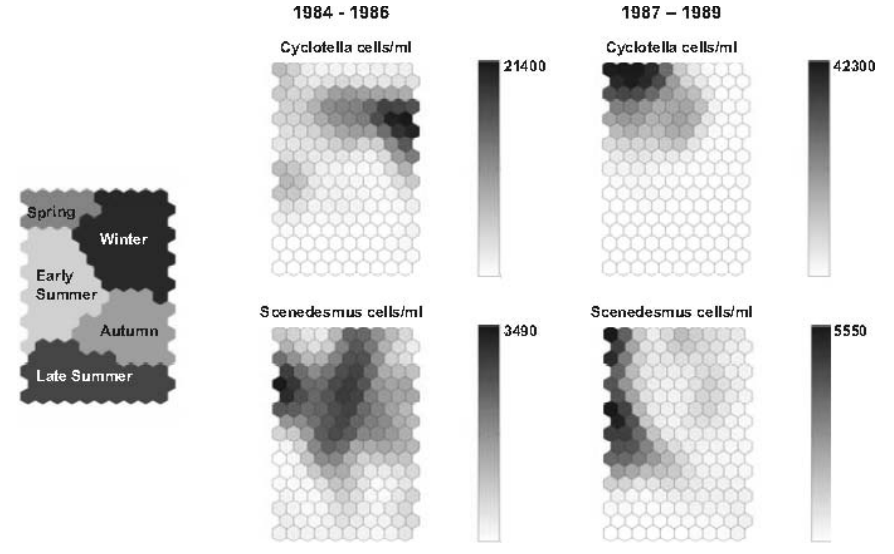


Fig. 16.13. Seasonal abundance clusters of the *Scenedesmus* and *Cyclotella* for two periods 1984 to 1986 and 1987 to 1989 with distinctive nutrient conditions in Lake Kasumigaura

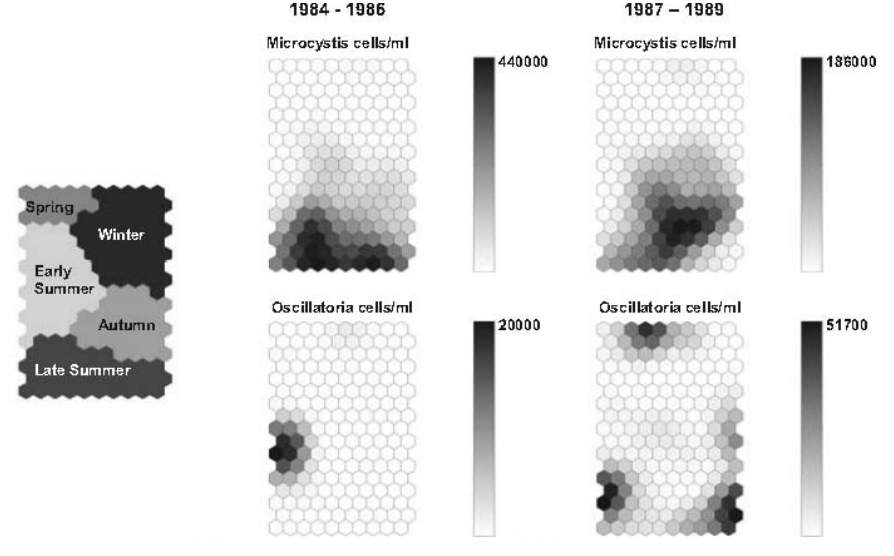


Fig. 16.14. Seasonal abundance clusters of the *Microcystis* and *Oscillatoria* for two periods 1984 to 1986 and 1987 to 1989 with distinctive nutrient conditions in Lake Kasumigaura

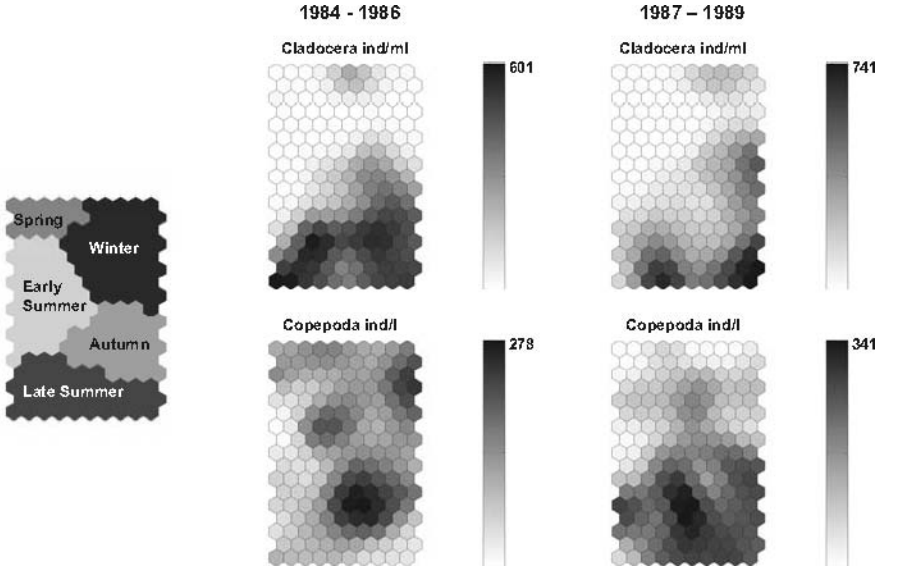


Fig. 16.15. Seasonal abundance clusters of cladocera and copepoda for two periods 1984 to 1986 and 1987 to 1989 with distinctive nutrient conditions in Lake Kasumigaura

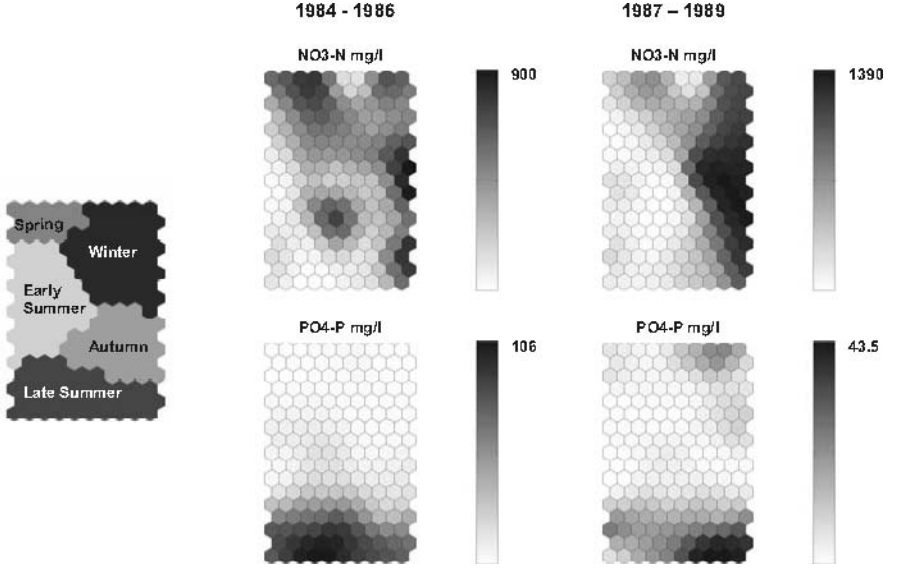


Fig. 16.16. Seasonal clusters of NO_3-N and PO_4-P concentrations for two periods 1984 to 1986 and 1987 to 1989 with distinctive nutrient conditions in Lake Kasumigaura

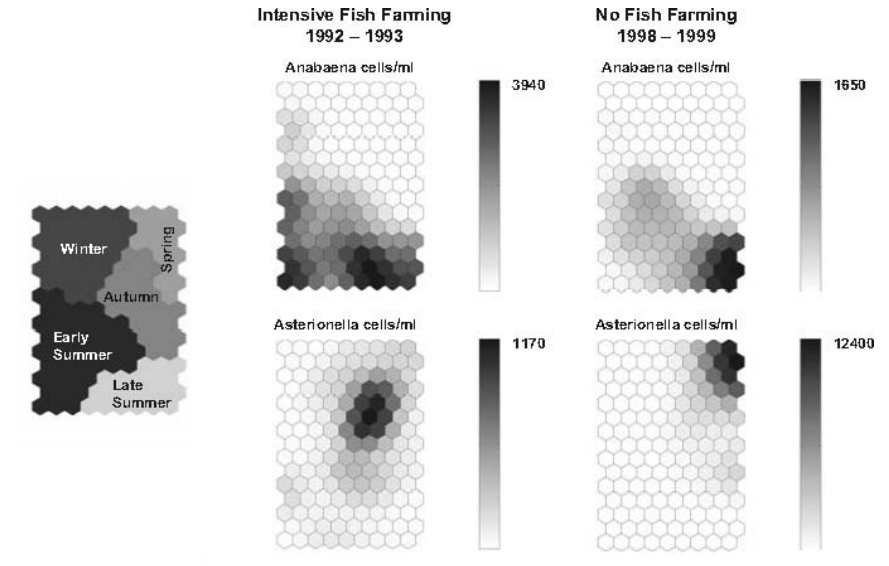


Fig. 16.17. Seasonal abundance clusters of *Anabaena* and *Asterionella* for two periods 1992 to 1993 and 1998 to 1999 with distinctive management conditions in Lake Soyang

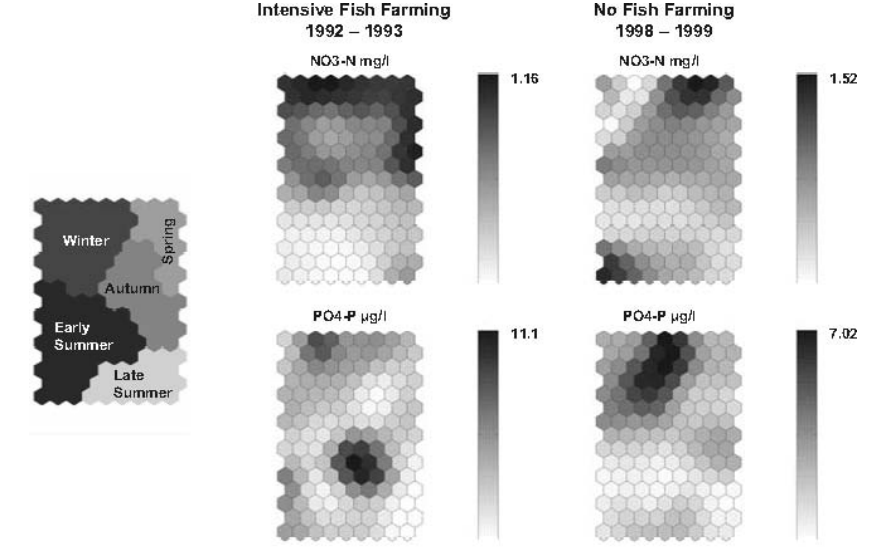


Fig. 16.18. Seasonal clusters of NO_3-N and PO_4-P concentrations for two periods 1992 to 1993 and 1998 to 1999 with distinctive management conditions in Lake Soyang

The Fig. 16.15 shows that abundances of both cladocera and copepoda for the two periods of Lake Kasumigaura are highest in early and late summer and partially in autumn with slightly increased numbers in period 2. The seasonal patterns of nutrient concentrations in Lake Kasumigaura in Fig. 16.16 show a 30% increase of NO_3-N from period 1 to 2 but at the same time a 30% decrease of PO_4-P. While NO_3-N peaks in winter in period 1 it peaks in spring and early summer in period 2. By contrast PO_4-P peaks in early and late summer in period 1 but peaks in winter in period 2.

The Fig. 16.17 shows seasonal abundance patterns of *Anabaena* and *Asterionella* during and after intensive fish farming in Lake Soyang. It indicates that the abundance of *Anabaena* peaked in early and late summer during fish farming but only in late summer and decreased by 50% after fish farming. By contrast *Asterionella* became most abundant in autumn during fish farming but peaked with in spring after fish farming at tenfold higher abundance.

The seasonal patterns of NO_3-N and PO_4-P were also visualised for the two different periods of management of lake Soyang in Fig. 16.18. The results demonstrate that NO_3-N concentrations were highest in winter and spring during fish farming but shifted to spring and early summer with 30% increased maxima. However highest PO_4-P concentrations occurred in winter and early summer during fish farming but only in winter at 30% lower magnitude after fish farming.

16.4 Discussion

16.4.1 Forecasting Seasonal Algal Abundances and Succession

The results in Fig. 16.4 have demonstrated that recurrent supervised ANN have the capacity for forecasting outbreaks and seasonal succession of different algal populations in distinctive lakes for one-week-ahead. Blue-green algae *Microcystis* which were observed to peak at 650,000 cells/ml in mid August 1986 in the shallow hypertrophic Lake Kasumigaura were forecasted to peak at 550,000 cells/ml in late August, and *Anabaena* which were observed to peak at 5,000 cells/ml in late August 1997 in the deep mesotrophic Lake Soyang were forecasted to peak at 6,000 cells/ml in early September. Diatoms *Cyclotella* that were observed to peak at 55,000 cells/ml in mid April 1986 in Lake Kasumigaura were forecasted to peak at 60,000 cells/ml in mid March, and *Asterionella* that were observed to peak at 4,000 cells/ml in mid May 1997 in Lake Soyang were forecasted to peak at 6,000 cells/ml in late April. The predicted timing was slightly delayed for outbreaks of the blue-green algal populations in the two lakes, but was up to 4 weeks earlier for the diatom populations.

Overall the results are encouraging towards the development of early warning systems for blue-green algal blooms based on real time forecasting by supervised

ANN. Such a system has been successfully tested for forecasting chlorophyll-*a* of a coastal bay (Lee, Fernando and Wong 2004) by means of on-line electronically measurable variables such as water temperature, dissolved oxygen, solar radiation.

16.4.2 Relationships between Algal Abundances, Seasons and Water Quality Changes

The results in Figs. 16.5 to 16.12 have shown that ordination and clustering by non-supervised ANN and sensitivity analyses by supervised ANN can be integrated to a powerful tool for analysing complex ecological relationships in data. It has revealed from data that blue-green algae and diatoms have distinctive relationships with water temperature, pH, NO_3-N- and PO_4-P- concentrations despite differences in the trophic state and morphometry of a lake. For both lakes clusters of the automatically mapped blue-green algae abundances corresponded well with the sensitivity curves indicating fastest growth at water temperatures higher then 20° C. By contrast clusters of the automatically mapped diatom abundances corresponded well with the sensitivity curves indicating fastest growth at water temperatures below 15° C namely between 9 and 12° C. These results comply with the temperature preferences postulated for specific algal assemblages (e.g. Reynolds 1984; Shapiro 1990), and discovered for *Microcystis* and *Stephanodiscus* in River Nakdong by Jeong, Recknagel and Joo (2003).

With regards to pH *Microcystis* in Lake Kasumigaura was mapped in the range between 8 and 9, and mapped together with *Anabaena* in Lake Soyang in the range between 7 to 8.5. As the pH of freshwater is determined by its CO_2 budget (Stumm and Morgan 1970) alkaline conditions are likely for a hypertrophic lake such as Kasumigaura as the availability of dissolved CO_2 can be seasonally limited when the primary productivity is highest (Schindler 1971). This may explain the slight upward shift of the pH range at which blue green algae predominately occur in Lake Kasumigaura compared to Lake Soyang. However the Figs. 16.7 and 16.8 clearly show that high abundances of blue-green algae in both lakes coincide with distinct alkaline conditions. Relationships between diatoms and pH revealed by Figs. 16.7 and 16.8 show that high abundances of *Cyclotella* in Lake Kasumigaura coincide with extreme high pH values greater than 9, whilst *Asterionella* in Lake Soyang seems to be most abundant at neutral pH.

Talling (1976) concluded from a series of experiments that both diatoms and blue-green algae appear to be photosynthesis tolerant regarding high pH and low CO_2 concentrations. Shapiro (1984) postulated that blue-green algae are physiologically adapted to cope well with low CO_2 concentrations and out-compete eukaryotic algae at high pH. Reynolds (1984) confirmed these findings for *Microcystis aeruginosa* and *Anabaena flos-aquae.* By virtue of the present results and the literature findings it can be concluded that *Microcystis* and *Anabaena* behave very much like *K*-selected species as they are specialised for

distinct environmental conditions such as higher water temperature and pH whereby *Cyclotella* and *Asterionella* tolerate a much broader range of conditions typical for *r*-selected species. As postulated by Reynolds (1984) the differentiation between *K*- and *r*-selected algal species relies very much on their capability to cope with sinking and grazing losses. Both *Microcystis* and *Anabaena* are considered *K*-selected as they minimise sinking losses by regulating their buoyancy and avoid grazing losses by forming large cell colonies or filaments as well as contain toxic substances (Reynolds 1984). By contrast *Cyclotella* and *Asterionella* are considered *r*-selected by having relatively high sinking losses because of their dense silica cell walls and being largely exposed to grazing (Reynolds 1984).

The elucidation of relationships between algal populations and nutrient conditions in lakes with different trophic states was another aim of the current study. With regards to NO_3-N concentrations clusters in Figs. 16.9 and 16.10 gave evidence that *Microcystis* and *Anabaena* reached highest abundances in both lakes when NO_3-N was lowest in concentration. On the other hand highest abundances of *Cyclotella* in Lake Kasumigaura and *Asterionella* in Lake Soyang were clustered at medium concentrations of NO_3-N. These findings were supported by corresponding sensitivity curves showing for both lakes that blue green algae peaked at low NO_3-N concentrations e.g. 0.4 mg/l in Lake Kasumigaura and 0.9 mg/l in Lake Soyang but diatoms peaked at higher NO_3-N concentrations e.g. 0.8 mg/l in Lake Kasumigaura and 1.3 mg/l in Lake Soyang. There are two possible explanations for these results: (1) blue-green algae diminish NO_3-N concentrations significantly by high NO_3-N uptake during maximum growth and diatoms are competitively excluded because of both non-favouring NO_3-N levels and water temperatures at that times, and (2) some blue-green algae out-compete diatoms at times of lowest NO_3-N concentrations by assimilating dissolved atmospheric nitrogen N_2 through heterocysts (Fay et al. 1968) as being clearly demonstrated for *Anabaena* but not yet for *Microcystis* (Reynolds 1984). The hypothesis (1) may explain best the results in Fig. 16.9 for Lake Kasumigaura whilst hypothesis (2) is more likely to reflect conditions in lake Soyang where summery stratification further accelerates sinking losses of diatoms. Opposite trends were revealed in the two non-P-limited lakes for relationships between blue-green algae, diatoms and PO_4-P in Figs. 16.11 and 16.12. Both *Microcystis* and *Anabaena* showed affinity to highest PO_4-P concentrations which read in Lake Kasumigaura 25 to 40 µg/l and in Lake Soyang 4 to 5 µg/l but *Cyclotella* and *Asterionella* became most abundant at low PO_4-P concentrations of 2.5 to 5 µg/l. The results for *Microcystis* and *Anabaena* correspond well with their intolerances of low PO_4-P concentrations postulated by Reynolds (1984). It was found by Mackereth (1953) that *Asterionella* cells have a phosphorus storage capacity equivalent to 24 times the absolute cell minimum that may explain *Asterionella*'s and *Cyclotella*'s tolerance to low PO_4-P concentrations as observed in this study. Although most findings of this research correspond well with current knowledge on algal specific relationships with pH and nutrient conditions it must be pointed out that these relationships are distinctively bi-directional as e.g. algal metabolism in turn changes nutrient and CO_2 budgets as well. Therefore patterns of

relationships between physical, chemical and biological properties of ecosystems automatically mapped from historical data by non-supervised ANN may reflect not necessarily the cause but the result of such relationships.

The results in Figs. 16.13 to 16.18 have demonstrated that ordination and clustering by non-supervised ANN can reveal seasonal and long-term patterns for ecological relationships in lakes.

The data for Lake Kasumigaura were seasonally ordinated and clustered for both the period 1 from 1984 to 1986 and the period 2 from 1987 to 1989 between which a significant increase of the TN (total nitrogen)/TP (total phosphorus) ratio from 10 to approximately 20 had been observed (Takamura et al. 1992). The resulting patterns revealed that *Cyclotella* peaked during winter in period 1 and with two times higher magnitude during spring in period 2. The shift of highest abundance of *Cyclotella* from winter to spring and period 1 to 2 seems to be determined by higher NO_3-N concentrations in period 2 and its know preference of NO3-N sufficiency discussed above for findings in Fig. 16.9. The green algae *Scenedesmus* appeared to be most dominant in spring and early summer in both periods with a 30% increased abundance in period 2. *Microcystis* clearly dominated in late summer in period 1 but shifted its dominance to autumn in period 2 with a 60% decrease in abundance. By contrast *Oscillatoria* shifted its dominance from early summer in period 1 to late summer in period 2 by more than doubling its maximum abundance. Results in Fig. 16.14 show competitive seasonal exclusion between *Microcystis* and *Oscillatoria* for both periods with a distinct PO_4-P limitation of *Microcystis* in autumn of period 2 (see Fig. 16.16) caused by high growth and PO_4-P consumption of *Oscillatoria* in late summer. While cladocera tended to have highest abundances in late summer of both periods, copepoda peaked at the transition from late summer to autumn. As in both periods high cladocera abundance coincided with abundant *Microcystis* there seems to be an indication for feeding of decaying *Microcycstis* cells by cladocera as observed in Lake Kasumigaura by Hanazato and Yasuno (1987) and Hanazato (1991). Results in Fig. 16.15 show also seasonal exclusion by predation of cladocera by copepoda for both periods.

The data for Lake Soyang were seasonally ordinated and clustered for both the period 1 from 1992 to 1993 with intensive fish farming and the period 2 from 1998 to 1999 with no fish farming. These periods were chosen as stopping fish farming in period 2 eased eutrophication of the lake by approximately 30% lower PO_4-P concentration (Kim *et al.* 2000). The seasonal clusters in Fig. 16.17 indicate a distinct response behaviour of blue-green algae and diatoms to the changed management between period 1 and 2. The abundance of *Anabaena* peaks in period 2 only in late summer at a 50% lower maximum but occurs in early summer at insignificant level only. One possible reason for this seasonal shift and reduced abundance of *Anabaena* can be found in Fig. 16.18 indicating that high PO_4-P concentrations in summer were typical for period 1 providing PO_4-P sufficiency as required by blue-green algae but occurred in winter only at 30% lower concentrations in period 2. On the other hand there is a shift in the dominance of *Asterionella* from autumn in period 1 to spring in period 2 with a ten times increaseed abundance (Fig. 16.17, bottom). Distinct spring blooms of

Asterionella in period 2 may have been triggered by a combination of the slightly increased NO_3-N concentrations in spring as a result of no fish farming, and its known PO_4-P storage capacity (Mackereth 1953) charged during the wintrily PO_4-P enrichment of the lake.

16.5 Conclusions

The current study has demonstrated that complex limnological time-series data can beneficially be processed by ANN in order to provide: (1) one-week-ahead forecasting of outbreaks of harmful algae or water quality changes by recurrent supervised ANN, and (2) clusters to unravel ecological relationships regarding seasons, water quality ranges and long-term environmental changes by non-supervised ANN. It has also been shown that these methods provide a useful framework for comparative studies between largely different lakes. Future work will focus on the integration of super- and non-supervised ANN into a representative lake data warehouse archiving long-term time-series of a broad range of lakes and rivers reflecting diverse climate, morphometric and eutrophic conditions. It will further facilitate "basic research on complex interactions (that) will lead to explanations for the variability and unpredictability that presently hamper lake management efforts…"
Carpenter (1988).

Acknowledgements

The authors are grateful to the Lake Kasumigaura Survey Group of the National Institute for Environmental Studies, Tsukua, Japan, for providing Lake Kasumigaura data.

References

Carpenter StR (ed.) (1988) Complex Interactions in Lake Communities. Springer-Verlag New York, Berlin

Chon TS, Young SP, Kyong HM, Eui YC (1996) Patternizing communities by using an artificial neural network. Ecological Modelling 90,69-78

Fay P, Stewart WD, Walsby AE, Fogg GE (1968) Is the heterocyst the site of nitrogen fixation in blue-green algae? Nature, 220, 810-812

Hanazato T, Yasuno M (1987). Evaluation of Microcystis as food for zooplankton in a eutrophic lake. Hydrobiologia 144, 251-259

Hanazato T (1991) Interrelationships between Microcystis and Cladocera in the highly eutrophic Lake Kasumigaura, Japan. Int. Revue ges. Hydrobiol. 76, 1, 21-36

Heo WM, Kim B (1997) The change of N/P ratio with eutrophication and Cyanobacterial blooms in Lake Soyang, Korea. Verh. Int. Verein. Limnol. 26, 491-495

Jeong KS, Recknagel F, Joo GJ (2003) Prediction and elucidation of population dynamics of the blue-green algae Microcystis aeruginosa and the diatom Stephanodiscus hantzschii in the Nakdong River-Reservoir System (South Korea) by a recurrent artificial neural network. In: Recknagel, F. (ed.) (2003) Ecological Informatics. Understanding Ecology by Biologically-Inspired Computation. Springer-Verlag, Berlin, 195 – 213

Kim B, Choi K, Kim C, Lee YH (2000) Effects of the summer monsoon on the distribution and loading of organic carbon in a deep reservoir, Lake Soyang, Korea. Water Research 34, 14, 3495-3504

Kim B (2002) Eutrophication of freshwater ecosystems in Korea, and the effect of monsoon. In: Lee, D. (ed). Ecology of Korea. Bumwo Publ., Seoul, 385-399

Kohonen T (1989) Self-Organization and Associative memory. Springer-Verlag, Berlin

Kohonen T (1995) Self-organising Maps. Springer-Verlag, Heidelberg

Lee JHW, Fernando TMKG, Wong KTM (2004) Real-time prediction of coastal algal blooms using artificial neural networks. Proc. of the 6th International Conference on Hydroinformatics, June 21-24, 2004, Singapore, Vol. 2, 1465-1472

Mackereth FJH (1953) Phosphorus utilisation of Asterionella formosa Hass. Journal of Experimental Botany 4, 296-313

Pineda F (1987) Generalisation of backpropagation to recurrent neural networks. Phys. Rev. Lett., 19, 59, 2229-2232

Recknagel F, Talib A, van der Molen D (2005) Phytoplankton community dynamics of two adjacent Dutch lakes in response to seasons and eutrophication control unravelled by non-supervised artificial neural networks. Ecological Modelling (in press)

Reynolds CS (1984) The Ecology of Freshwater Phytoplankton. Cambridge University Press, Cambridge

Shapiro J (1990) Current beliefs regarding dominance of by blue-greens: the case for the importance of CO2 and pH. Verh.Int.Verein.Limnol. 24, 38-54

Stumm W, Morgan JJ (1970) Aquatic Chemistry. Wiley, New York

Takamura N, Otsuki A, Aizaki M, Nojiri Y (1992) Phytoplankton species shift accompanied by transition from nitrogen dependence to phosphorus dependence of primary production in Lake Kasumigaura, Japan. Archive Hydrobiol. 124, 129-148

Talling JF (1976) The depletion of carbon dioxide from lake water by phytoplankton. Journal of Ecology 64, 79-121

Van Tongeren OFR, van Liere L, Gulati RD, Postema G, Bosewinkel PJ (1992) Multivariate analysis of the plankton communities in the Loosdrecht lakes: relationship with chemical and physical environment. Hydrobiologia 233, 105-117

Varis O (1991) A canonical approach to diagnostic and predictive modelling of phytoplankton communities. Arch.Hydrobiol. 122,147-166

Varis O, Sirvia H, Kettunen J (1989) Multivariate analysis of lake phytoplankton and environmental factors. Arch.Hydrobiol. 117,163-175

Walter M, Recknagel F, Carpenter C, Bormans M (2001) Predicting eutrophication effects in the Burrinjuck Reservoir (Australia) by means of the deterministic model SALMO and the recurrent neural network model ANNA. Ecological Modelling 146, 1-3, 97-114

Chapter 17

Hybrid Evolutionary Algorithm* for Rule Set Discovery in Time-Series Data to Forecast and Explain Algal Population Dynamics in Two Lakes Different in Morphometry and Eutrophication

H. Cao · F. Recknagel · B. Kim · N. Takamura

17.1 Introduction

It has been demonstrated that ecological time series, which are highly complex and nonlinear can be successfully unraveled and predicted by artificial neural networks (ANN) and genetic algorithms (e.g. Recknagel et al. 1997; Recknagel 1997; Recknagel et al. 1998; Maier et al. 1998; Jeong et al. 2001; Whigham and Recknagel 2001; Wilson and Recknagel 2003; Stockwell 1999; Recknagel et al. 2002; Jeong et al. 2003; Lee, Fernando and Wong 2004). Even though ANN are very competitive in classifying or predicting noisy data by minimizing the root mean square error of approximations they lack an explicit representation. By contrast, Whigham and Recknagel (2001) proposed grammar based genetic programming to evolve functions and rules, and Bobbin and Recknagel (2003) applied an evolutionary algorithm to discover predictive rules for population dynamics in limnological data. Even though both approaches allowed to discover predictive rules for ecological relationships they had following limitations: (1) the rules were relatively simple with attributes being associated only with constant parameters rather than functions to reflect complex relationship between multiple attributes, and (2) the parameters which determine the output values on the rules were generated randomly rather than being simultaneously optimised during the evolution. Whigham and Recknagel (2001) performed the hill climbing mutation for the fine-tuning of the random real numbers and Bobbin and Recknagel (2003) adopted a self-adapting evolutionary algorithm to modify these parameters. However both methods fail when the number of parameters increases with the complexity of the rule.

This research aims at rule-based prediction and explanation of population dynamics of diatom and blue-green algae species in the two different lakes Kasumigaura and Soyang by means of a hybrid evolutionary algorithm (HEA). HEA evolves the structure of the rule set by using genetic programming, and

* Please find Demo Version of the Hybrid Evolutionary Algorithm on CD enclosed in the book.

optimises the random parameters in the rule set by using a general genetic algorithm. Rules discovered by HEA have the IF-THEN-ELSE structure and allow imbedding complex functions synthesised from various predefined arithmetic operators. The maximum tree depth and rule set size control the complexity of rule sets.

The results demonstrate that HEA allows to discover rule sets which predict well unseen data and represent causal relationships between physical and chemical variables and algal population dynamics.

17.2 Materials and Methods

17.2.1 Study Sites and Data

Lake Kasumigaura is situated in the southeastern part of Japan and receives flow from 56 rivers and streams. Its catchment area of 2135 km^2 consists of paddy areas but is largely urbanised and industrialised. The lake was turned from a brackish into a freshwater lake 5 years after a floodgate to the Pacific Ocean was implemented in 1963.

Lake Soyang is situated in the northeastern part of South Korea and fed by the Soyang River contributing 90% of the inflowing water. Nutrient loadings to Lake Soyang are predominantly caused by non-point sources from paddy and forest areas, and temporarily by in-lake fish farming using net cages.

Tab. 17.1 summarises characteristics of the two lakes. Tab. 17.2 provides details of the limnological databases of the two lakes.

Data of Lake Kasumigaura were collected with a column sampler of 2m at the centre of the Takahamairi Bay. Data of Lake Soyang were collected in meter steps at the central station and averaged over the upper 10 m for the present study. As the measurement intervals of the raw data from both lakes were highly irregular and sampling dates different for physical, chemical and biological data the data were interpolated to create consistent daily values as required for the development of rule set models.

Tab. 17.1. General characteristics of Lake Kasumigaura and Lake Soyang

	Lake Kasumigaura	Lake Soyang
Surface area km^2	219.9	45
Maximum volume km^3	662	2900
Maximum depth m	7	110
Mean depth m	3.9	42
Water residence time years	0.55	0.7
Catchment area km^2	1597	2675
Circulation type	non-stratified	warm monomictic

Tab. 17.2. Limnological properties reflected by the databases of Lake Kasumigaura and Lake Soyang

Limnological Variables	Lake Kasumigaura (1984 – 1993)	Lake Soyang (1990 – 2000)
	Mean / Min / Max	Mean / Min / Max
PO_4 μg/l	14.16 / 1 / 235	3.6 / 0.15 / 19.75
NO_3 mg/l	0.52 / 0.001 / 2.39	1.11 / 0.4 / 2.2
Si mg/l	3.29 / 0.015 / 12.49	
Chl*a* μg/l	74.5 / 0.69 / 279.5	3.4 / 0.18 / 46.1
Turbidity NTU (Turb)		1.43 / 0.4 / 10.35
Secchi Depth m (SD)	0.85 / 0.25 / 3.8	4.17 / 0.6 / 10
pH	8.75 / 7.12 / 10.13	7.3 / 6.2 / 9.1
Water Temperature °C (WT)	16.37 / 2.1 / 32	15.1 / 4.2 / 29.4
Solar Radiation Jcm^{-2} day^{-1}	1281 / 65 / 3364	420 / 54 / 2869
Phytoplankton cells/ml *Microcystis* *Cyclotella* *Anabaena* *Asterionella*	 38637 / 1 / 644117 5160 / 1 / 75420	 1304 / 1 / 34393 681 / 1 / 20594

17.2.2 Hybrid Evolutionary Algorithm

Evolutionary algorithms (EA) are adaptive methods which mimic processes of biological evolution, natural selection and genetic variation. They search for suitable representations of a problem solution by means of genetic operators and the principle of "survival of the fittest". Due to their merits of self-organization, self-learning, intrinsic parallelism and generality, EA have been successfully applied to pattern recognition, economic prediction, optimum control and parallel processing (Goldberg 1989; Bäck et al. 1997).

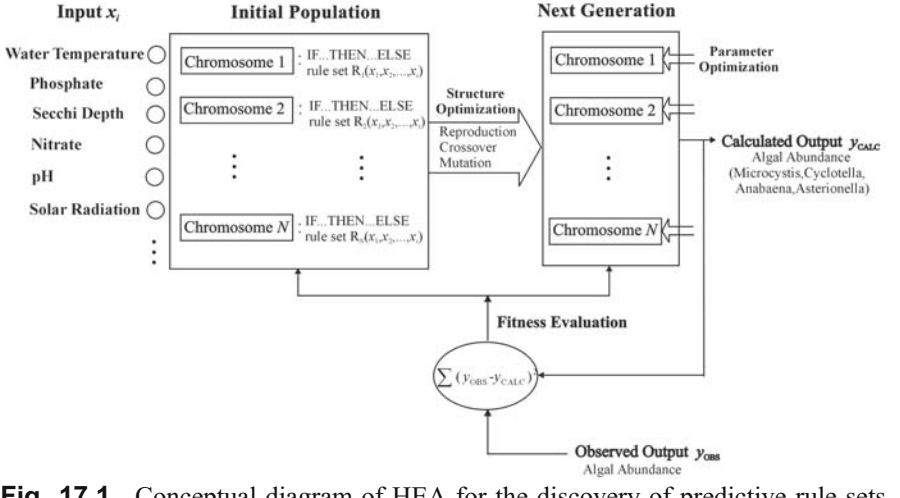


Fig. 17.1. Conceptual diagram of HEA for the discovery of predictive rule sets in water quality time-series

The principal framework of the rule discovery in water quality time-series by the suggested hybrid evolutionary algorithm (HEA) is represented in Fig. 17.1. The detailed algorithm for the rule discovery and parameter optimization by HEA is shown in Fig. 17.2.

HEA uses genetic programming (GP) to generate and optimize the structure of rule sets and a genetic algorithm (GA) to optimize the parameters of a rule set. GP (Koza 1992, 1994; Banzhaf et al. 1997) is an extension of genetic algorithms (GA) (Holland 1975; Mitchell 1996) in which the genetic population consists of computer programs of varying sizes and shapes. In standard GP, computer programs can be represented as parse trees, where a branch node represents an element from a function set (arithmetic operators, logic operators, elementary functions of at least one argument), and a leaf node represents an element from a terminal set (variables, constants and functions of no arguments). These symbolic programs are subsequently evaluated by means of "fitness cases". Fitter programs are selected for recombination to create the next generation by using genetic operators, such as crossover and mutation. This step is iterated for consecutive generations until the termination criterion of the run has been satisfied. A general genetic algorithm (GA) is used to optimize the random parameters in the rule set. We give the detailed descriptions of HEA in following sections.

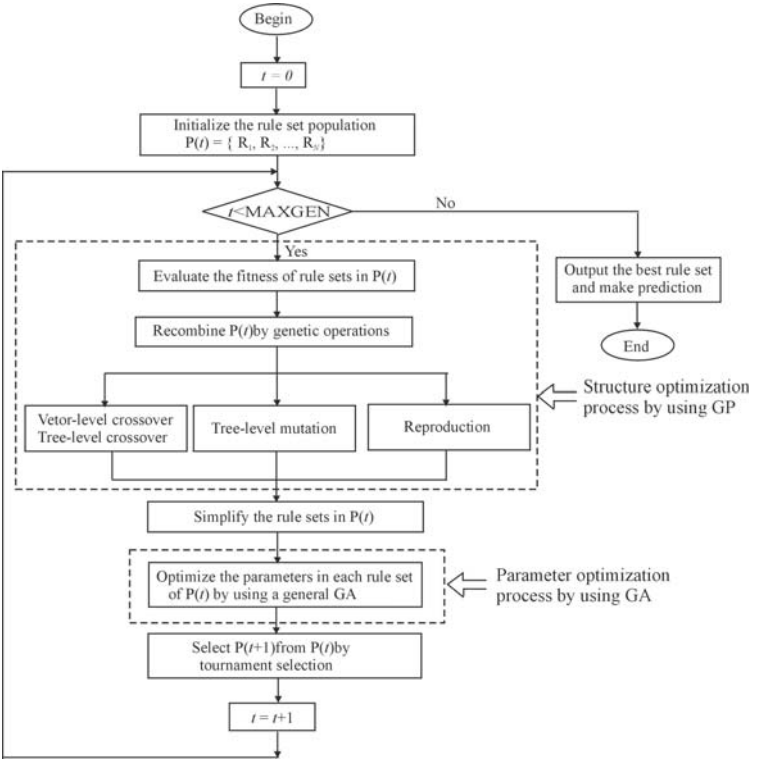


Fig. 17.2. Flowchart of the hybrid evolutionary algorithm (HEA)

17.2.2.1 Structure Optimization of Rule Sets Using GP

Encoding

We suppose each rule set has the form of

$$\begin{aligned}
&\text{IF } (T_{IF1})\\
&\text{THEN } y = T_{THEN1}\\
&\text{ELSE IF } (T_{IF2})\\
&\quad\text{THEN } y = T_{THEN2}\\
&\quad\quad\vdots\\
&\quad\quad\text{ELSE IF } (T_{IFK})\\
&\quad\quad\quad\text{THEN } y = T_{THENK}\\
&\quad\quad\quad\text{ELSE } y = T_{ELSEK+1}
\end{aligned} \tag{17.1}$$

where K is the size of the rule set, i.e. the number of IF-branches, y is the output variable. Then each chromosome in the rule set population can be represented as a vector of binary trees denoted as (T_{IF1}, T_{THEN1}, T_{IF2}, T_{THEN2}, …, T_{IFK}, T_{THENK}, $T_{ELSEK+1}$).

By defining the following three function sets as

Logic function set: $F_L = \{\text{AND, OR}\}$

Comparison function set: $F_C = \{>, <, \geq, \leq\}$

Arithmetic function set: $F_A = \{+, -, *, /, \sin, \cos, \exp, \ln\}$

The function sets of the IF_Tree (i.e. T_{IF1}, T_{IF2}, …, T_{IFK}} and the THEN/ELSE_Tree (i.e. T_{THEN1}, T_{THEN2}, …, T_{THENK}, $T_{ELSEK+1}$) can be described as

$$F_{IF} = F_L \cup F_C \cup F_A \quad \text{and} \quad F_{THEN/ELSE} = F_A$$

respectively. The terminal sets of the IF_Tree and the THEN/ELSE_Tree are the same as

$$T = \{x_1, \ldots, x_n, c\}$$

where n is the number of input variables and c is a random constant. For example, a rule set with the form of

$$\begin{aligned}
&\text{IF } ((\ln(|PO_4|) < 98) \text{ AND } ((WT > 30.8) \text{ OR } (pH * SD \leq 49.4)))\\
&\text{THEN } \textit{Microcystis} = WT + pH * \sin(Chla) - NO_3 * PO_4\\
&\text{ELSE IF } ((WT>20.5) \text{ AND } (PO_4/SD \leq 4))\\
&\quad\text{THEN } \textit{Microcystis} = pH * \exp(WT) + Chla/4\\
&\quad\text{ELSE } \textit{Microcystis} = (NO_3\text{-}3.5)/(PO_4 * WT + 4.8)
\end{aligned} \tag{17.2}$$

can be represented as a vector of binary trees ((T_{IF1}, T_{THEN1}, T_{IF2}, T_{THEN2}, T_{ELSE3}) illustrated in Fig. 17.3. Besides the function sets, the complexity of a rule set can be controlled by the predefined maximum size of a rule set (MAXK) and the maximum tree depth (D_{IF} and $D_{THEN/ELSE}$ for the IF_Tree and the THEN/ELSE_Tree respectively).

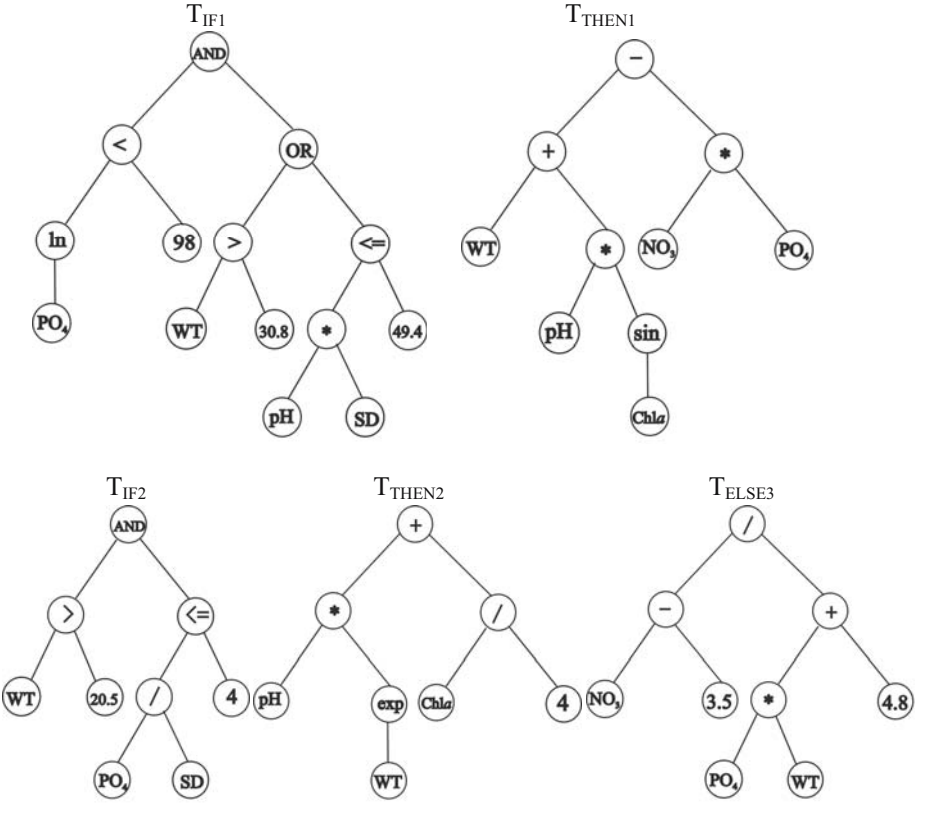


Fig. 17.3 An example of the representation of a rule set in GP

Fitness Evaluation

Suppose that the *i*th observed data for the input variables and the output variable are (x_{1i}, x_{2i}, …, x_{ni}) and y_i respectively. As for each rule set with the form of (1), we calculate the return values (TRUE/FALSE) of T_{IF1}, T_{IF2}, …, T_{IFK} in sequence based on the observed values of input variables to find out which condition is first satisfied. Say the first IF_Tree to be satisfied is T_{IFm}, we choose the corresponding THEN_Tree T_{THENm} to calculate the predicted value of y_i denoted as $\hat{y}_i$. If none of these IF_Trees is satisfied, the only choice is to use the last tree $T_{ELSEK+1}$ to calculate $\hat{y}_i$. Such procedure is performed on each data point from the training data. We define the RMSE (Root Mean Square Error) as the fitness function:

$$\text{Fitness} = \sqrt{\frac{1}{k}\sum_{i=1}^{k}(\hat{y}_i - y_i)^2}$$

where k is the number of training data points. Obviously, here the lower the fitness value is, the better is the rule set.

Genetic Operators

Since each rule set is represented as a vector of trees, there are two levels of crossover available, the vector-level crossover and the tree-level crossover.

Consider two parents:

Parent a: $(T_{IF1}^{(a)}, T_{THEN1}^{(a)}, T_{IF2}^{(a)}, T_{THEN2}^{(a)}, \ldots, T_{IFKA}^{(a)}, T_{THENKA}^{(a)}, T_{ELSEKA+1}^{(a)})$

Parent b: $(T_{IF1}^{(b)}, T_{THEN1}^{(b)}, T_{IF2}^{(b)}, T_{THEN2}^{(b)}, \ldots, T_{IFKB}^{(b)}, T_{THENKB}^{(b)}, T_{ELSEKB+1}^{(b)})$

where KA and KB are the sizes of the rule set a and b respectively.

The vector-level crossover is performed as follows. Randomly select a position between the pairs of IF_THEN statement within Parent a and Parent b as the crossover point, say j and k for a and b respectively. Then swap the corresponding IF-THEN-ELSE statements below the crossover points and produce two new rule sets. We use either of them as the crossover offspring on condition that its size does not exceed MAXK. Fig. 17.4 illustrates such procedure of doing vector-level crossover.

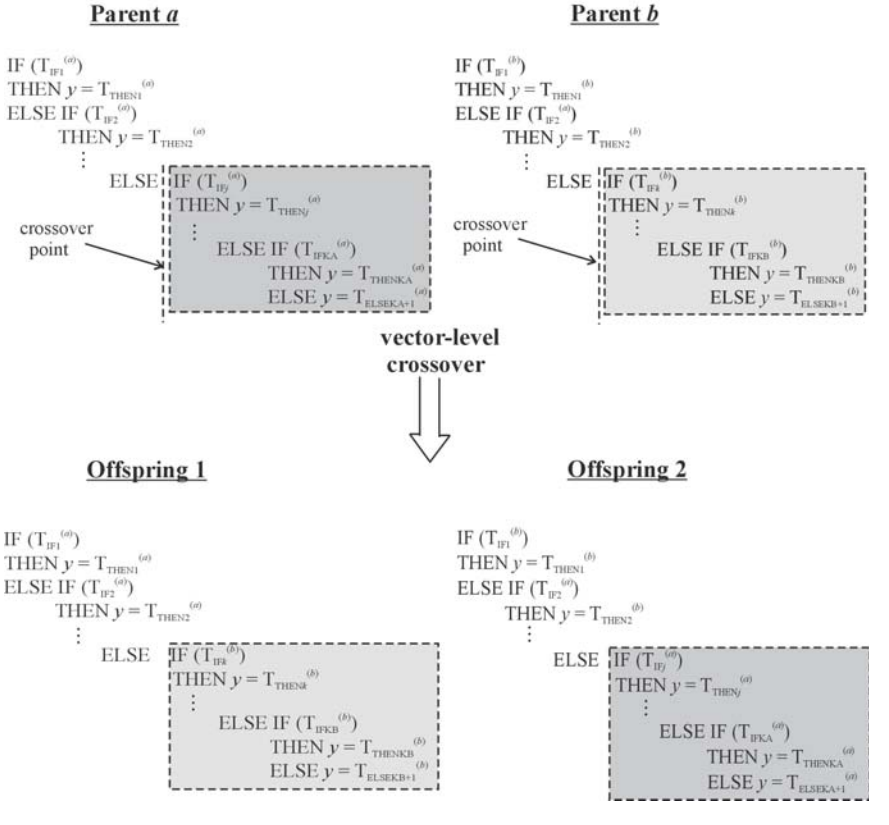
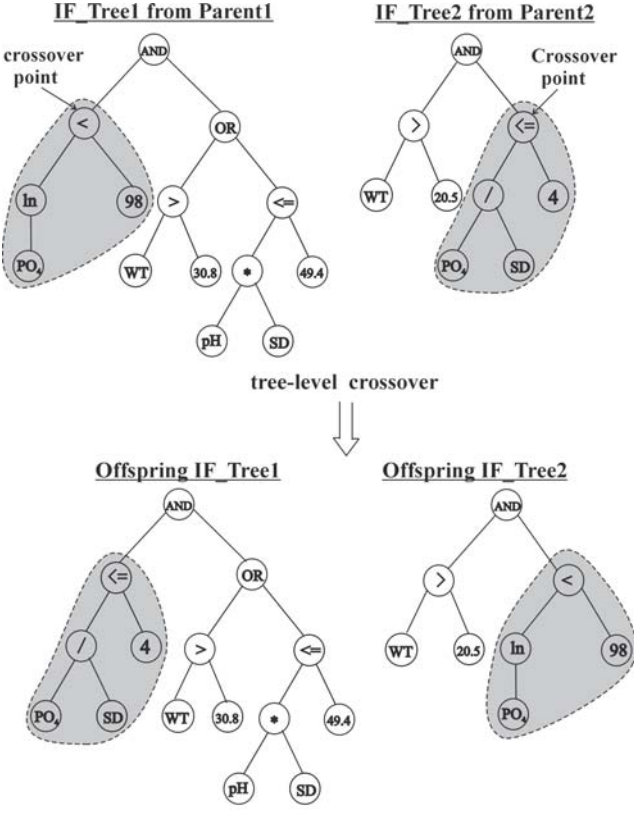


Fig. 17.4. Illustrations of vector-level crossover of rule sets

The tree-level crossover is performed between the IF_Trees and the THEN/ELSE_Trees of two parents in sequence. First we do the IF_Tree crossover as follows. Randomly choose an IF_Tree from each parent and a node within the tree as a crossover point as well, swap the subtrees rooted at the crossover points and produce two new trees, then use either of them as the corresponding IF_Tree of the offspring on condition that its maximum depth does not exceed D_{IF}. Fig. 17.5 illustrates an example of tree-level IF_Tree crossover. It needs to be pointed out that in the IF_Tree there are three different types of function nodes which come from F_L, $F_{C,}$ F_A respectively, to ensure that the crossover always produces legal rule sets, and only the same type of nodes are selected as the crossover points. Afterwards the THEN/ELSE Tree crossover is similarly done as described



Notes:

IF_Tree1 from Parent1: IF $((\ln(|PO_4|) < 98)$ AND $((WT > 30.8)$ OR $(pH * SD \leq 49.4)))$

IF_Tree2 from Parent2: IF $((WT > 20.5)$ AND $(PO_4/SD \leq 4))$

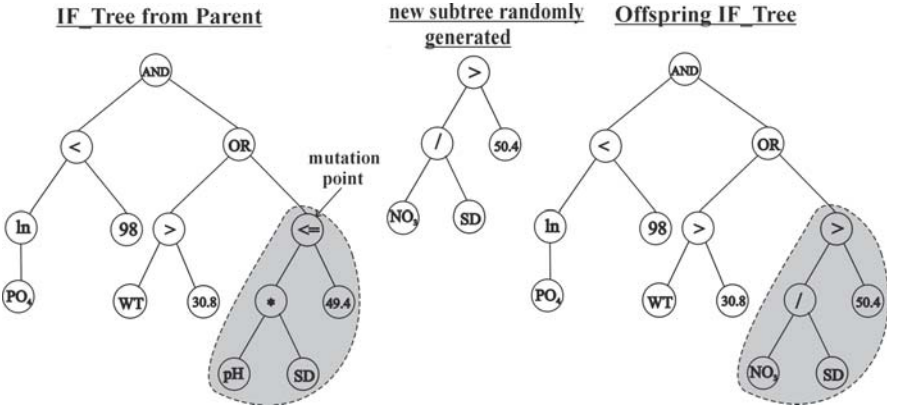
Offspring IF_Tree1: IF $((PO_4/SD \leq 4)$ AND $((WT > 30.8)$ OR $(pH * SD \leq 49.4)))$

Offspring IF_Tree2: IF $((WT > 20.5)$ AND $(\ln(|PO_4|) < 98))$

Fig. 17.5. Illustrations of tree-level IF_Tree crossover of rule sets

above. The only difference is that we can choose any node as the crossover point due to their identical arithmetic node type. Finally we select either of the parents and replace the IF_Tree and the THEN/ELSE_Tree chosen previously with the newly generated ones by the above two-step crossover. Thus we get a complete crossover offspring of the two parents.

Similarly the tree-level mutation is performed on the IF_Tree and the THEN/ELSE_Tree of one parent in sequence. The tree-level mutation of IF_Tree begins by randomly selecting an IF_Tree from the parent and also a node within the tree as the mutation point, replacing the subtree rooted at the mutation point with a randomly generated new subtree, thus producing an offspring IF_Tree. Fig. 17.6 illustrates an example of tree-level IF_Tree mutation. Afterwards the mutation of the THEN/ELSE Tree is in a similar way as discussed above. Thus we get a complete mutation offspring of the parent by replacing the IF_Tree and the THEN/ELSE_Tree chosen previously with the newly generated ones by the above two-step mutation.



Notes:

IF_Tree from Parent: IF $((\ln(|PO_4|) < 98)$ AND $((WT > 30.8)$ OR $(pH * SD \leq 49.4)))$

Offspring IF_Tree: IF $((\ln(|PO_4|) < 98)$ AND $((WT > 30.8)$ OR $(NO_3 / SD > 50.4)))$

Fig. 17.6. Illustrations of tree-level IF_Tree mutation of rule sets

Simplification of Rule Sets

The simplification of rule sets includes the simplification of the IF_Tree and the THEN/ELSE Tree. We use the following consecutive steps to simplify the IF_Tree in each rule set:

(1) Simplification of the arithmetic subtrees: It is done by replacing subtrees which consist of arithmetic operations in F_A between constants by their calculated values.

(2) Simplification of the comparison subtrees: It is done by replacing the subtrees which consist of comparison operations in F_C between constants by their comparison outcome, i.e. 0 or 1 for TRUE and FALSE respectively.

(3) Simplification of the logic subtrees: We use Tab. 17.3 to simplify the AND subtrees and OR subtrees which consist of 0 or 1 in their branch nodes.

Tab. 17.3. The simplification of the logic subtrees

AND	OR
0 AND 0 = 0	0 OR 0 = 0
0 AND 1 = 0	0 OR 1 = 1
1 AND 0 = 0	1 OR 0 = 1
1 AND 1 = 1	1 OR 1 = 1
0 AND subtree = 0	0 OR subtree = subtree
1 AND subtree = subtree	1 OR subtree = 1

We only use the above step (1) to simplify the THEN/ELSE_Tree in each rule set. In addition we delete the redundant pairs of IF_THEN statements from the original rule set by checking the number of the input data points which can satisfy the condition of the IF_Tree. If the number is zero, then the corresponding IF-THEN pair is regarded as making no sense and should be deleted from the original rule set. The size of the rule set can thus be significantly reduced in this way.

The simplification of rule set is performed on all individuals in every generation. This procedure should be done prior to the parameter optimization because it is helpful to reduce the total number of parameters to optimize while maintaining the fitness of the rule set.

17.2.2.2 Parameter Optimization of Rule Sets Using a General Genetic Algorithm

As the parameters in the rule set, especially those contained in the IF-Trees, play an important role in calculating the accuracy of the rule set, they need to be optimised in each generation. Here we design a general genetic algorithm (GA) to approach this task.

GAs can have various forms due to different representations, fitness evaluations and genetic operators which may vary with specific problems. Among all these components, genetic operators, including crossover and mutation, are usually considered as the most important parts. Here we used a novel crossover operator based on the nonconvex linear combination of multiple parents during the recombination of the population, which proved to work stably and effectively in solving the problem of multiple parameters optimization (Yu et al. 1999).

Encoding

At the beginning, we first check all the constants contained in the IF_Trees and the THEN/ELSE_Trees of the rule set, including counting the number of constants l and recording their positions. Each individual in the parameter population can then be represented as an l-dimensional row vector $(c_1, c_2, \ldots, c_l)$ where each

component c_i for i =1, 2, …, l is encoded as a floating number and generated randomly ranging from 0 to 20 during the initialization of the parameter population.

Fitness Evaluation
Before the fitness evaluation of an individual in the parameter population, we first return to the original rule set and replace all constants with the corresponding components of the row vector (i.e. the individual) and then follow the same procedure as in Section 17.2.2.1.2 to calculate the fitness.

Genetic Operators
We use a multiple-parent crossover operator to create a new individual in the parameter population in the following way. Randomly select M different individuals from the old population (M>2) denoted as X_1, X_2, ⋯, X_M where X_k = (c_{1k}, c_{2k}, …, c_{lk}) (k:1~M). Produce M coefficients α_k , where α_k ranges from a to b (a<0, b>1), which satisfy $\sum_{k=1}^{M}\alpha_k = 1$. Generate a new individual, X, by the nonconvex linear combination of these M individuals as follows:

$$X = \sum_{k=1}^{M}\alpha_k X_k$$

If the fitness value of X is lower than that of the worst individual in the current population, then replace it with X. This step is iterated a predetermined maximum number (MAX) of times. There are three adjustable control parameters M, a, b in this procedure. Setting their optimal values depends upon the properties of the specific problem.

Selection Strategy
We use tournament selection with sample size of 4 to recombine the new rule set population. That is, each time we randomly choose 4 different individuals from the current rule set population and compare their fitness values. The best one among them is added to the new population. This procedure is repeated until the predefined population size N is reached. In the meantime an elitism strategy is adopted which means we always keep the best rule set in the current generation to the next generation.

17.2.2.3 Forecasting by Rule Sets

Once the best rule set is obtained in one run, we then test its validity and generality by calculating the predicted values on the testing data points and the RMSE for the testing data. A lower RMSE for the unseen data usually implies that the rule set has better generalised the patterns found in the training data.

17.3 Case Studies Lake Kasumigaura and Lake Soyang

17.3.1 Parameter Settings and Measures

To examine the effectiveness of HEA, we applied it to predict 7-days-ahead seasonal succession of *Microcystis* and *Cyclotella* for Lake Kasumigaura, and of *Anabaena* and *Asterionella* for Lake Soyang. For Lake Kasumigaura the daily input data for PO_4, NO_3, Secchi depth, pH, water temperature, solar radiation, Chl*a*, and Si as well as daily output data for M*icrocystis* and *Cyclotella* of the years 1984 to 1985 and 1987 to 1993 were used for training, and the data of 1986 were used for testing the generalisation behaviour of the resulting rule sets. For Lake Soyang the daily input data for PO_4, NO_3, Secchi depth, pH, water temperature, solar radiation, Chl*a*, and Turbidity as well as daily output data for *Anabaena* and *Asterionella* of the years 1990 to 1996 and 1998 to 2000 were used for training, and the data of 1997 were used for testing.

100 runs were conducted independently for each data set. All the experiments were performed on a Hydra supercomputer (IBM eServer 1350 Linux) with a peak speed of 1.2 TFlops by using the programming language C. The parameter settings of HEA are listed in Tab. 17.4.

Tab. 17.4. Parameter settings of the hybrid evolutionary algorithm for rule set discovery

Structure Optimization (GP)	$N = 200$ $F_L = \{AND, OR\}$ $F_C = \{ >, <, \geq, \leq \}$ $F_A = \{ +, -, *, /, \exp, \ln \}$ $MAXK = 4$ $D_{IF} = D_{THEN/ELSE} = 4$ $MAXGEN = 100$
Parameter Optimization (GA)	popsize = 50 $a = -0.5$ $b = 1.5$ $M = 8$ $MAX = 500$

In order to validate the results of different rule sets not only the training error (fitness) but also the testing error (RMSE) is calculated as follows:

$$\sqrt{\frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2}$$

where m is the number of testing data points, y_i and $\hat{y}_i$ are the ith observed value and the ith predicted value of the output variable such as *Microcystis* abundance.

17.3.2 Results and Discussion

Tab. 17.5 shows the best rule sets in terms of the minimal testing error in 100 runs with 7-days-lagged input data for each algal population. Fig. 17.7 shows the validation results for the best rule sets.

Tab. 17.5. The best rule sets in terms of the minimal testing error obtained in 100 runs with 7-days-lagged input data for each algal population

Algal Population	Best Rule Sets	Condition	Training Error	Testing Error
Microcystis	**RULE SET 1:** IF (PO_4 < 52.11) THEN *Microcystis* = WT*(WT-14.05)+ PO_4*WT ELSE *Microcystis* = Chla*(pH*15.32+WT-152.37)	*Mic1* *Mic2*	512.95	392.47
Cyclotella	**RULE SET 2:** IF ($PO_4 \leq$ 171.54 AND Si*exp(PO_4) > 85.04) THEN *Cyclotella* = 3.67 ELSE IF (WT ≥ 16.36) THEN *Cyclotella* = 335.95 ELSE IF ((PO_4*exp(PO_4)) ≥ 126.71) THEN *Cyclotella* = exp(pH)/pH+exp(pH)/3.96 ELSE IF ((Si*exp(WT)) ≥ 74.60) THEN *Cyclotella* = exp(pH)/pH+254.678 ELSE *Cyclotella* = 335.95	*Cyc1* *Cyc2* *Cyc3* *Cyc4* *Cyc5*	923.17	863.07
Anabaena	**RULE SET 3:** IF (WT > 20.66) THEN *Anabaena* = (Turb*70.73-27.41)* 11.93*ln(|Chla |)/pH ELSE *Anabaena* = 37.27*Chla*Chla /pH	*Ana1* *Ana2*	193.14	73.57
Asterionella	**RULE SET 4:** IF ((exp(SD) ≥ 55.95)OR(WT > 17.30)) THEN *Asterionella* = 18.34 ELSE IF ((WT > 12.88)OR(PO_4 ≥ 4.7)) THEN *Asterionella* = 120.56 ELSE IF (SD*exp(SD) < 37.65) THEN *Asterionella* = Turb*(exp(pH)/pH)* Chla ELSE *Asterionella* = Turb*Chla*59.83	*Ast1* *Ast2* *Ast3* *Ast4*	115.47	67.71

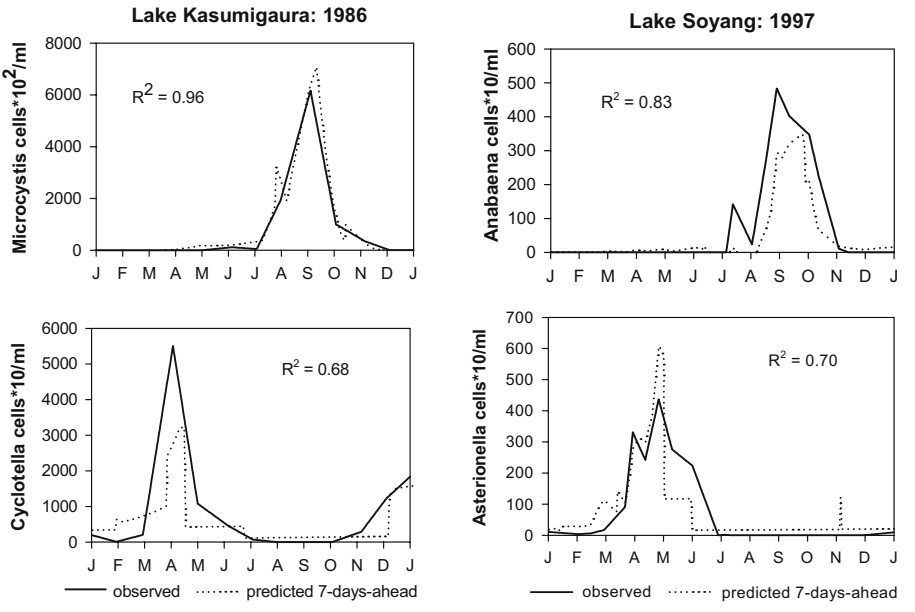


Fig. 17.7. 7-days-ahead forecasting of *Microcystis* and *Cyclotella* for Lake Kasumigaura in 1986 (left column), and of *Anabaena* and *Asterionella* for Lake Soyang in 1997 (right column) by using the rule sets shown in Tab. 17.5.

The predicted timing and magnitude of the summer peak of *Microcystis* (Fig. 17.7, top, left) for Lake Kasumigaura correspond very well with the measured data ($R^2 = 0.96$). The predicted timings of the spring and autumn peaks of *Cyclotella* (Fig. 17.7, bottom, left) compare well with the observed data, but the magnitude is slightly under-estimated resulting in a $R^2 = 0.68$.

As for Lake Soyang, the forecasting result of the summer peak of *Anabaena* in 1997 (Fig. 17.7, top, right) is reasonable regarding both magnitude and timing with a $R^2 = 0.83$ even though the observed slight early summer peak was missed. The predicted timings of two spring peaks of *Asterionella* in 1997 (Fig. 17.7, bottom, right) are almost consistent with the measured data, but the magnitude of the second peak is slightly over-estimated resulting in a $R^2 = 0.70$.

From the above results it can be concluded that the forecasting of the diatom populations *Cyclotella* and *Asterionella* is more challenging compared to the blue-green algal populations *Microcystis* and *Anabaena*. This finding is also reflected by the complexity of rule sets discovered by HEA that is higher for diatoms than for blue-green algae. While RULE SET 1 for *Microcystis* and RULE SET 3 for *Anabaena* in Tab. 17.5 are relatively simple consisting of one single rule and their IF conditions are associated with only one input variable respectively. On the contrary, RULE SET 2 for *Cyclotella* and RULE SET 4 for *Asterionella* are much more complicated in the structure which consists of 4 and 3 IF condition branches respectively.

To further analyse and interpret the rule sets in Tab. 17.5 discovered by HEA, we highlighted the activation of rule condition branches as well as interesting regions within the forecasting periods for each algal population.

Fig. 17.8 illustrates the 7-days-ahead forecasting of *Microcystis* segmented by THEN-branch and ELSE-branch of RULE SET 1. As the IF condition is only related to the input variable PO_4, we draw the curve of input PO_4 for Lake Kasumigaura in 1986 and mark the threshold value as well. Obviously Fig. 17.8 reflects the preference of *Microcystis* to relatively high concentrations of PO_4. We shadow the region which satisfies the ELSE condition branch i.e. when $PO_4 \geq 52.11$. However as indicated by the THEN-branch of RULE SET 1 algal abundance of *Microcystis* at low PO_4 determined only by the combination of both PO_4 concentration and water temperature. This result is consistent with the findings by Reynolds (1984) that *Microcystis* is intolerant to both low PO_4 concentrations and water temperature.

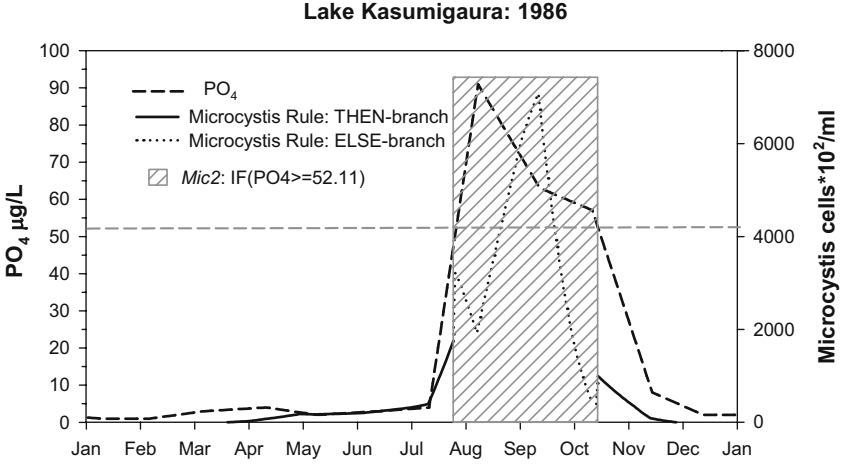


Fig. 17.8. 7-days-ahead forecasting of *Microcystis* for Lake Kasumigaura in 1986 segmented by THEN-branch and ELSE-branch of RULE SET 1 in Tab. 17.5.

Fig. 17.9 shows the results of a sensitivity analysis regarding the THEN- and ELSE-branches of RULE SET 1. It reflects once more the distinct sensitivity of *Microcystis* to PO_4 and water temperature of the THEN-branch (Fig. 17.9, left). By contrast if the condition *Mic2* is satisfied i.e. $PO_4 \geq 52.11$, Fig. 17.9 (right) indicates that *Microcystis* is very sensitive to high temperature and pH value causing highest abundance of *Microcystis*. The value ranges of pH and WT indicate that the high *Microcystis* abundances coincide with pH values higher than 8 and water temperature over 20°C. Both values comply with literature findings based on field observations and laboratory experiments (Reynolds 1984; Shapiro 1990).

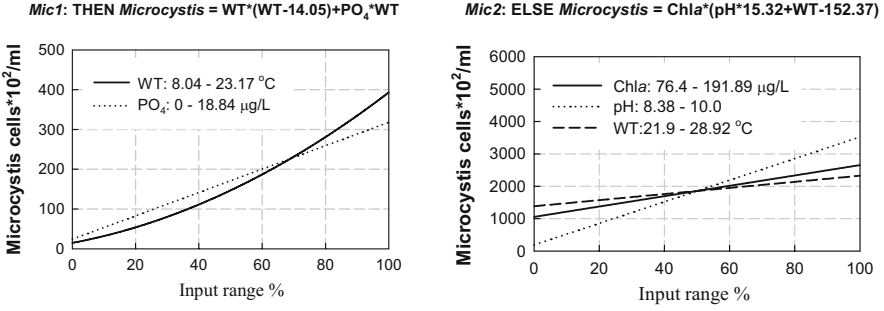


Fig. 17.9. Sensitivity analysis with disturbance ±STDEV of input data for THEN-branch (left) and ELSE-branch (right) of RULE SET 1 for *Microcystis* in Tab. 17.5.

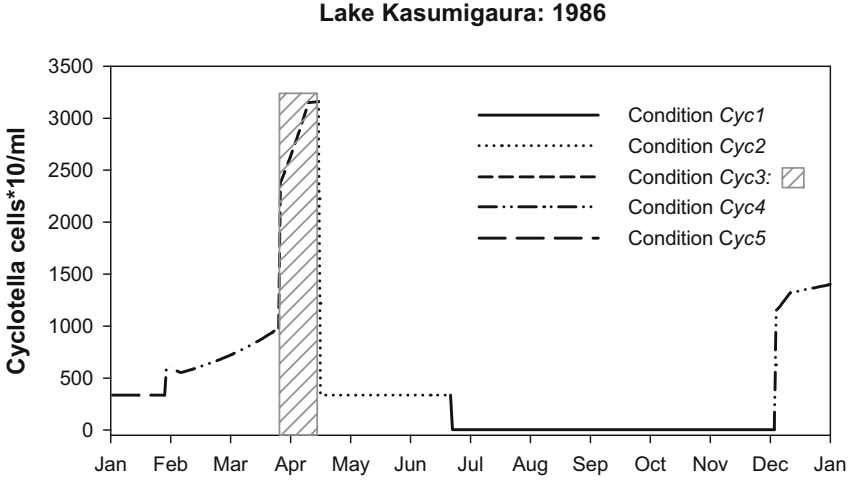


Fig. 17.10. 7-days-ahead forecasting of *Cyclotella* for Lake Kasumigaura in 1986 segmented by different condition branches of RULE SET 2 in Tab. 17.5.

Fig. 17.10 illustrates the 7-days-ahead forecasting of *Cyclotella* segmented by different condition branches of RULE SET 2. As the structure of RULE SET 2 is rather complex we focus our interpretation at condition *Cyc3* that determines highest abundances of *Cyclotella*. The condition *Cyc3* can be rewritten as follows:

Cyc3: IF $((PO_4 > 171.54$ OR $Si*\exp(PO_4) \leq 85.04)$ AND $PO_4 \geq 3.57$ AND $WT < 16.36)$

where the condition $PO_4*\exp(PO_4) \geq 126.71$ in RULE SET 2 is equivalent to $PO_4 \geq 3.57$. The condition *Cyc3* clearly demonstrates that *Cyclotella* favours water temperature below 16°C and low PO_4 concentrations above 3.57µg/l. These results correspond well with literature findings that diatom cells are tolerant to low water temperatures (e.g. Reynolds 1984) and can have a phosphorus storage

capacity that may explain their tolerance to low PO_4-P concentrations Mackereth (1953).

As the outputs of other three conditions are constant, we only plot the sensitivity curves for *Cyc3* and *Cyc4* in Fig. 17.11. It can be seen that in both cases *Cyclotella* experiences high sensitivity to the change of pH and a higher pH gives rise to a higher algal abundance. In fact as observed in the measured data, *Cyclotella* in Lake Kasumigaura peaks at extreme high pH values of 9 to 9.3.

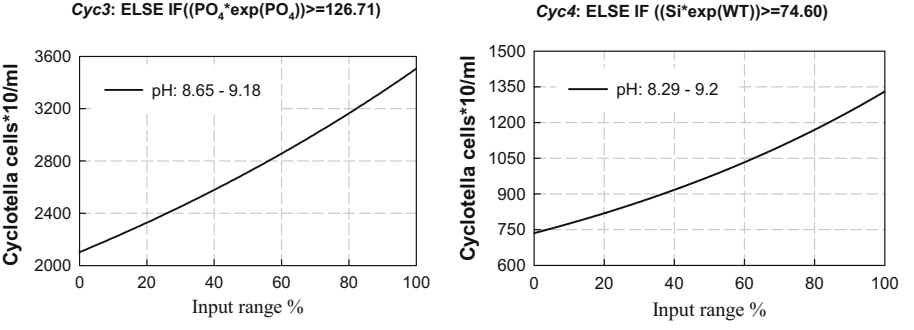


Fig. 17.11. Sensitivity analysis with disturbance ±STDEV of input data for the *Cyc3* (left) and *Cyc4* (right) condition branches of RULE SET 2 for *Cyclotella* in Tab. 17.5.

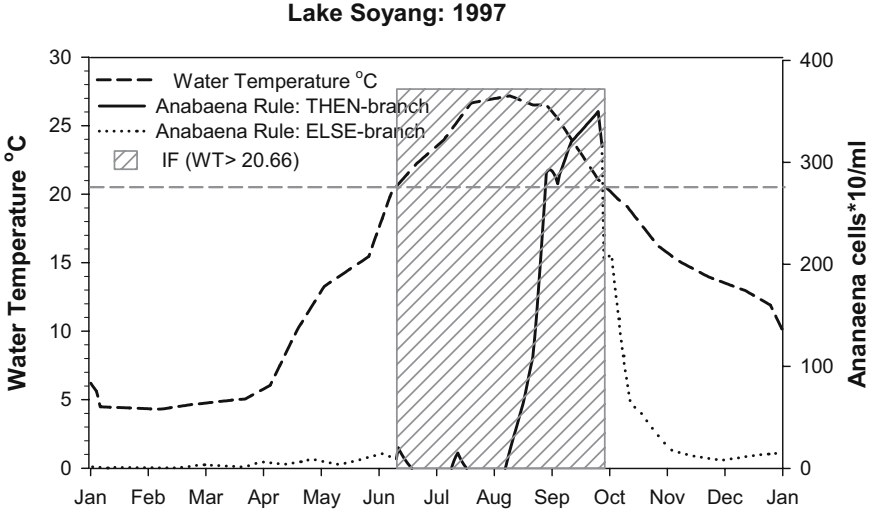

Lake Soyang: 1997
Water Temperature °C
Anabaena Rule: THEN-branch
Anabaena Rule: ELSE-branch
IF (WT> 20.66)
Water Temperature °C
Ananaena cells*10/ml
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec Jan

Fig. 17.12. 7-days-ahead forecasting of *Anabaena* for Lake Soyang in 1997 segmented by THEN-branch and ELSE-branch of RULE SET 3 in Tab. 17.5.

Fig. 17.12 illustrates the 7-days-ahead forecasting of *Anabaena* segmented by the THEN- and ELSE-branches of RULE SET 3. We also plot the curve of water temperature for Lake Soyang in 1997 which is the only input variable related to

the condition and shadow the region which satisfies the condition WT>20.66. It can be seen from the graph that *Anabaena* peaks at a water temperature higher than 20°C. However the specific abundance of *Anabaena* is ultimately determined by the combined effect of water temperature, turbidity, pH and Chl*a* as reflected by THEN-branch of the rule set. That is the reason why the shadowed window shows a quite a wide abundance range of *Anabaena* during June and July.

Fig. 17.13 shows the sensitivity analysis regarding the THEN- and ELSE-branchs of RULE SET 3. Both cases show that *Anabaena* experiences little sensitivity to changes of pH, and high sensitivity to Chl*a*. When the condition *Ana1* (i.e. WT>20.66) is satisfied, the sensitivity of *Anabaena* is also high to turbidity. Both relationships are typical for the period after the summer monsoon in Lake Soyang that has been observed to coincide with high abundance of *Anabaena* (Kim *et al.* 2000).

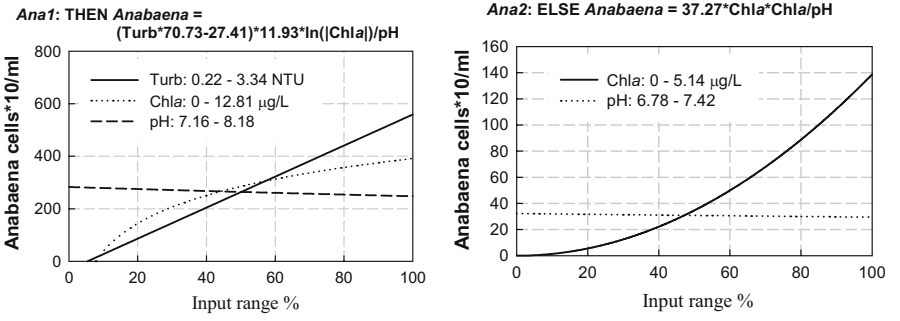


Fig. 17.13. Sensitivity analysis with disturbance ±STDEV of input data for THEN-branch (left) and ELSE-branch (right) of RULE SET 3 for *Anabaena* in Tab. 17.5.

Fig. 17.14 illustrates the 7-days-ahead forecasting of *Asterionella* segmented by different condition branches of RULE SET 4. In order to interpret the RULE SET 4 we focus at the condition *Ast3* that reflects conditions for highest abundances of *Asterionella*. The condition *Ast3* reads as follows:

Ast3: IF (WT ≤ 12.88 AND PO_4<4.7 AND SD<2.65)

It is notable that the condition SD*exp(SD)<37.65 in RULE SET 4 is equivalent to SD<2.65 and the condition exp(SD)≥55.95 in RULE SET 4 is equivalent to SD≥4.02. It becomes obvious that *Asterionella* prefers an environment with water temperatures below 13°C, PO_4 concentration below 4.7µg/l and Secchi depths lower than 3m. This findings confirm once more the tolerance of diatoms to low water temperatures and PO_4 concentrations as discussed before for *Asterionella* in Lake Kasumigaura.

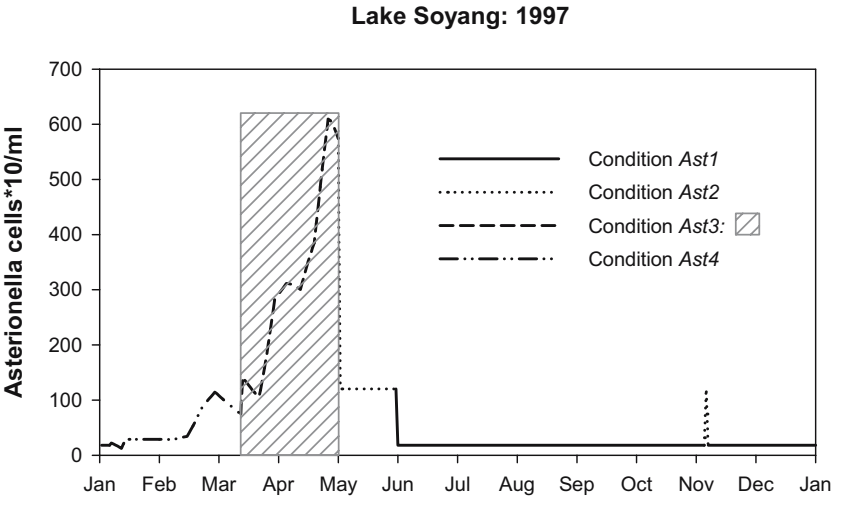


Fig. 17.14. 7-days-ahead forecasting of *Asterionella* for Lake Soyang in 1997 segmented by different condition branches of RULE SET 4 in Tab. 17.5.

As the outputs of RULE SET 4 for the other two conditions are constant, we only plot the sensitivity curves for *Ast3* and *Ast4* in Fig. 17.15. It can be seen that in both cases the sensitivity of *Asterionella* is high to the changes of turbidity and Chl*a* whose increases will cause a higher abundance of *Asterionella*.

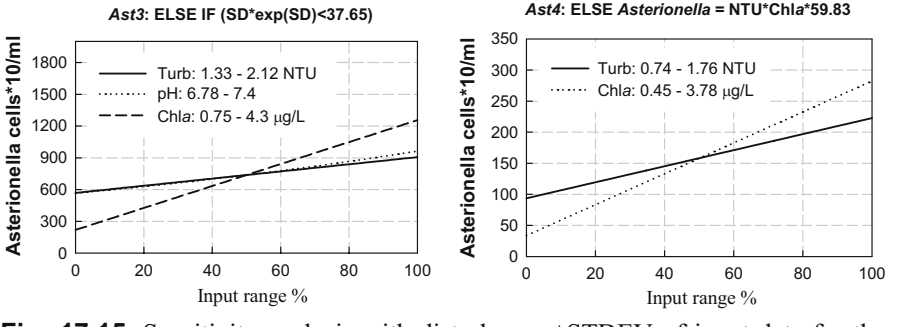


Fig. 17.15. Sensitivity analysis with disturbance ±STDEV of input data for the *Ast3* (left) and *Ast4* (right) condition branches of RULE SET 4 for *Asterionella* in Tab. 17.5.

17.4 Conclusions

A hybrid evolutionary algorithm (HEA) has been developed to discover predictive rule sets in complex ecological data. It has been designed to evolve the structure of rule sets by using genetic programming and to optimise the random parameters in the rule sets by means of a genetic algorithm.

HEA was successfully applied to long-term monitoring data of the shallow, eutrophic Lake Kasumigaura (Japan) and the deep, mesotrophic Lake Soyang (Korea). The results have demonstrated that HEA is able to discover rule sets, which can forecast for 7-days-ahead seasonal abundances of blue-green algae and diatom populations in the two lakes with relatively high accuracy but are also explanatory for relationships between physical, chemical variables and the abundances of algal populations. The explanations and the sensitivity analysis for the best rule sets correspond well with theoretical hypotheses and experimental findings in previous studies.

References

Bäck T, Hammel U, Schwefel H-P (1997) Evolutionary computation: comments on the history and current state. IEEE Transaction on Evolutionary Computation. 1(1), 5-16

Banzhaf W, Nordin P, Keller RE, Francone FD (1997) Genetic Programming: An Introduction on the Automatic Evolution of Computer Programs and its Applications. Morgan Kaufmann

Bobbin J, Recknagel F (2003) Evolving rules for the prediction and explanation of blue-green algal succession in lakes by evolutionary computation. In: Recknagel, F. (ed.), 2003. *Ecological Informatics. Understanding Ecology by Biologically-Inspired Computation*. Springer-Verlag Berlin, Heidelberg, New York, 291-310

Goldberg DE (1989) Genetic Algorithms in Search, Optimization and Machine Learning. Reading, MA: Addison Welsey

Holland JH (1975) Adaptation in Natural and Artificial System, Ann Arbor, MI: University of Michigan Press

Jeong KS, Joo GJ, Kim HW, Ha K, Recknagel F (2001) Prediction and elucidation of phytoplankton dynamics in the Nakdong River (Korea) by means of a recurrent artificial neural network. Ecological Modelling. 146, 115-129

Jeong KS, Kim DK, Whigham P, Joo GJ, 2003. Modeling *Microcystis aeruginosa* bloom dynamics in the Nakdong River by means of evolutionary computation and statistical approach. Ecological Moddelling. 161, 67-78

Kim B, Choi K, Kim C, Lee YH (2000) Effects of the summer monsoon on the distribution and loading of organic carbon in a deep reservoir, Lake Soyang, Korea. Water Research 34, 14, 3495-3504

Koza JR (1992) Genetic programming: On the Programming of Computers by Means of Natural Selection. Cambridge, MA: MIT Press.

Koza, J R (1994) Genetic programming II: Automatic Discovery of Reusable Programs. Cambridge, MA: MIT Press

Liu Y, Yao X (1999) Time series prediction by using negatively correlated neural networks. Lecture Notes in Artificial Intelligence. vol. 1585. Springer-Verlag, Berlin, pp. 325-332

Lee JHW, Fernando TMKG, Wong KTM (2004) Real time prediction of coastal algal blooms using artificial neural networks. Proceedings of the 6th International Conference on Hydroinformatics

Mackereth FJH (1953) Phosphorus utilisation of Asterionella formosa Hass. Journal of Experimental Botany 4, 296-313

Maier HR, Dandy GC, Burch MD (1998) Use of artificial neural networks for modeling cyanobacteria *anabaena* spp. in the River Murray, South Australia. Ecol. Model. 105, 257-272

Mitchell M (1996) An Introduction to Genetic Algorithms. Cambridge, MA: MIT Press

Recknagel F, French M, Harkonen P, Yabunaka K (1997) Artificial neural network approach for modeling and prediction of algal blooms. Ecological Moddelling. 96, 1-3, 11-28

Recknagel F (1997) ANNA - Artificial neural network model predicting species abundance and succession of blue-green Algae. Hydrobiologia, 349, 47-57

Recknagel F, Fukushima T, Hanazato T, Takamura N, Wilson H (1998) Modelling and prediction of phyto- and zooplankton dynamics in Lake Kasumiguara by artificial neural networks. Lakes and Reservoirs: Research and Management. 3, 123-133

Recknagel F, Bobbin J, Whigham P, Wilson H (2002) Comparative application of artificial neural networks and genetic algorithms for multivariate time-series modelling of algal blooms in freshwater lakes. Journal of Hydroinformatics 4, 2, 125-134

Reynolds CS (1984) The Ecology of Freshwater Phytoplankton. Cambridge University Press, Cambridge

Shapiro J (1990) Current beliefs regarding dominance of by blue-greens: the case for the importance of CO2 and pH. Verh.Int.Verein.Limnol. 24, 38-54

Stockwell DRB (1999) Machine learning methods for ecological modelling, Chapter Genetic Algorithms II: Species distribution modeling. Kluwer Academic Publishers. pp. 123-144

Whigham PA, Recknagel F (2001) Predicting Chl.a in freshwater lakes by hybridising process-based models and genetic algorithms. Ecological Modelling. 146, 243-251

Whigham PA, Recknagel F (2001) An inductive approach to ecological time series modelling by evolutionary computation. Ecological Modelling. 146, 275-287

Wilson H, Recknagel F (2003) A generic artificial neural network model for dynamic predictions of algal abundance in freshwater lakes. In: Recknagel, F. (ed.), 2003. Ecological Informatics. Understanding Ecology by Biologically-Inspired Computation. Springer-Verlag Berlin, Heidelberg, New York, 265-287

Yu JX, Cao HQ, Chen YY, Kang LS Yang HX (1999) A new approach to estimation of the electrocrystallization parameters. Journal of Electroanalytical Chemistry, 474(1), 69-73

Chapter 18

Multivariate Time Series Prediction of Marine Zooplankton by Artificial Neural Networks

C.H. Reick · A. Grünewald · B. Page

18.1 Introduction

Most applications of Neural Networks are based on their high *adaptivity* to almost any set of given input-output relations ("patterns"). An example is pattern recognition: here, a usually complex input, e.g. a picture of a number, has to be mapped on a much simpler output, e.g. the binary representation of that number. Networks trained to reproduce such input-output relations can be used to identify a complex input from the much simpler output.

In applications to time series prediction these input-output relations relate past to future data. But here the task is not to reproduce a previously known input-output relation. Instead one wants the Neural Network to produce correctly from known past data presently unknown future data. Here another feature of Neural Networks comes into play, namely their ability to *generalize*. Neural Networks cannot only be trained to learn particular input-output relations. During training they also seem to develop a more general representation of these relations. On the basis of these relations predictions can be performed. The nature of this more general representation is not very clear and may depend on the chosen network structure, but one can understand it as a kind of inter- or extrapolation of the trained input-output relations. Operationally the ability of a Neural Network to generalize is usually defined as its ability to produce correct outputs — i.e. predictions in the present context — for inputs that have not been used during training (see e.g. Hertz et al. (1991)). It is clear that an appropriate generalization may fail, but there are striking examples where this generalization works extremely well, e.g. for the prediction of the (secondary) protein structure from its sequence of amino acids (Rost and Sander 1993) or the prediction of chaotic dynamics (Wan 1994).

Especially when low quality data are used for prediction or if the number of simultaneous variables is high, it is often hard to judge, whether a particular

Neural Net is able to generalize. This situation is often encountered when working with environmental data sets, as everyone knows, who tried to work e.g. with biological time series (see e.g. Reick and Page (2000)). The reasons for this low quality are simple: First, environmental data are typically taken under non-laboratory conditions, i.e. external disturbances cannot be controlled and the data get noisy. Second, one has usually to measure what one can get, and not what one would like to measure. So one cannot be sure, whether the data are representative for some hidden deterministic dynamics. And finally, long measurement campaigns are expensive so that environmental data sets are typically quite short compared to their noise level. This problem is only insufficiently compensated by measuring simultaneously several variables (the extra costs are typically low), because hereby one can only improve the information on particular system states, but cannot gain additional information on the diversity of system states; this could only be obtained from sufficiently long time series. But this information on the diversity of states is indispensable for predictions of high quality. Moreover, the information of additional variables is often redundant and also noisy, to the consequence that by using these data as additional inputs in Neural Nets their performance can get even worse.

The prediction quality is usually measured by computing the average prediction error for a number of prediction instants. But the question is, how far this prediction error can be trusted, when a Neural Network is used to predict unknown data. When working with Neural Networks one experiences that the lower the data quality, the less reliable are the computed prediction errors. As already discussed, this situation is especially encountered, when working with environmental data so that here one should always carefully analyze their reliability. This means one has to inquire the ability of a Neural Network to generalize. How to do this by crossvalidation techniques will be discussed in section II. Unfortunately, crossvalidation is very laborous, because the same Neural Network has to be trained over and over again with different parts of the available data. The solution can only be a complete automatization of the training process. Standard Neural Network software, like e.g. SNNS (Zell 1994), supports mainly the visual supervision of the training at the computer monitor. But this is much too laborous when performing crossvalidation studies. Alternatively, one could use the programming interfaces, that are part of many Neural Network products. But besides the uncomfortability of such a solution, there is a more fundamental problem with automatization: Many training algorithms have been developed in the past and many of them are available in Neural Network packages. But when using them for automatized training the main problem is how to stop the training, such that the network is neither under- and nor overadapted in order to guarantee optimal generalization. For visual supervision at the screen, there is a widely accepted stopping technique by Weigend et al. (1991). As a first step to automatization we show in section III how this technique can be cast into an algorithm. Finally we show in section IV how crossvalidation and automatized training can be applied to an environmental data set, namely to plankton time series from the North Sea. In contrast to most other prediction studies, we will not

show how well Neural Nets predict these time series, but instead show how the failure of their ability to generalize can be substantiated.

18.2 Generalization

To find a Neural Network, by which a particular time series can be correctly predicted, many prediction experiments have to be performed. In these experiments one varies the type of input data, modifies the preprocessing of the data and changes the internal structure of the Neural Networks. The success of a particular prediction experiment is usually measured by the prediction error that is obtained when trying to predict data that were not used during training. The Neural Network with the least prediction error will then be chosen to perform actual predictions.

When working with data of high quality, this procedure often works quite well. But when working with poor data the prediction quality is low, and this leads to two problems. First, the experimental effort to find a Neural Network with an acceptable prediction error increases. Whereas this is mainly a practical problem, the second is of more fundamental nature: The reliability of the predictions gets questionable. In the case of a Neural Network with a small prediction error (small in relation to a characteristic scale of the data; the error computed for available data, not in actual predictions) a doubling or even tripling of the error would still be a small error. Therefore, a small prediction error is a good indication that for actual predictions the error will also be small. For low quality data the situation is different. If one finds only networks with prediction errors that cannot be judged small, e.g. 20-30% relative error, then a doubling or tripling of the error in actual predictions will no more be acceptable. Therefore, in this case, the reliability of the predictions gets problematic.

In our opinion, this problem can be discussed in the context of the more general problem of generalization. The term ”generalization” is usually bound to a small output error (here: prediction error) for non-training data. It is clear that a small prediction error is an indication of a successful generalization. But what about generalization, if the error is not small? Even in that case a Neural Network may correctly reproduce essential features of a time series — although with a significant error. Obviously in such a situation other aspects than the prediction error get relevant for the question of generalization, as e.g. the reliability of the prediction error, as discussed above. Therefore, in the following, we will discuss how a generalization success or failure can be detected for low quality predictions.

We start the discussion by distinguishing several causes for predictive performance failures of Neural Networks:

Unpredictability. The data may not represent a phenomenon governed by a common rule. It is obvious that in such a case any prediction method will fail.

Despite its triviality, this point is mentioned here, because Neural Networks are often tentatively applied to phenomena whose predictability is questionable, like in the case of stock returns.

Poor data. It may happen that in principle the considered phenomenon is predictable, but the data used for training contain not enough information. Several types of information deficiencies can be distinguished: First, there may be simply not enough data available. Second, the data may be too noisy. Third, the data set may be incomplete, in the sense that certain aspects of the phenomenon are not represented by the data. And finally, the data may contain the wrong information. This can happen if the phenomenon to be predicted is recently governed by a different rule as before (instationarity). If the data used for training contain no or only partial information on the present rule, the Neural Network performs the predictions according to the past rule, and consequently fails.

Overadaptation. Training a Neural Network means to change its parameters iteratively until it reproduces a given set of input-output relations with sufficient accuracy. This accuracy is measured by the error between the intended and the actual output. Except for the initial training phase, where strong fluctuations may be observed, this error usually decreases monotonically during training. But when the error is monitored for a different data set, that is not used for training, one often observes that beyond a certain point the error increases. This indicates that a good adaptation and a good generalization are conflicting aims. So, when employing Neural Networks for predictive purposes, one has to take care that the training process is terminated before an overadaptation occurs. This problem, especially how to detect and prevent an overadaptation, will be discussed in more detail in section III.

Underadaptation. As already mentioned, after an initial phase typically not only the error for the training data decreases, but also the error for independent data. So, terminating the training too early may lead to an underadaptation and a reduced ability to distinguish different system states. This problem is easily circumvented by sufficiently long training periods. More fundamental is the problem, that the training process may stick to a local minimum in the error landscape so that, although the Neural Network may in principle be capable of a good adaptation, the optimal parameters are not found during training. This problem is well known from numerical mathematics and no general solution exists (see e.g. Press et al. (1986)).

Unsuited network structure. It is a general experience that when changing the various structural elements of a Neural Network, like the number of neurons, the topology of their connections or the type of activation and output functions, its predictive performance changes. Unfortunately there are no general rules what type of network structure is appropriate for a particular problem so that one can never exclude that a performance failure is the result of an unsuited network type. A partial solution to this problem is the use of training procedures that change not only the network parameters, but also its topology.

From these five causes for performance failures the first two (”unpredictability” and ”poor data”) are independent of Neural Networks and

from the remaining three only ”overadaptation” and ”unsuited network structure” are related to generalization failures, while the cause ”underadaptation” is the result of an insufficient adaptation. Nevertheless, in practice, one is usually not able to identify the particular cause for a performance failure so that in practice a general performance failure cannot be distinguished from a generalization failure.

But besides prediction quality, there are also a second and third aspect of generalization and these turn out to be measurable almost independently of the prediction quality. The second aspect, as already mentioned, concerns the *reliability* of the predictions. One characteristic of generalization failures is their independence from the data: Let us assume, we had trained a Neural Network with high quality data. We now take other data, documenting the same phenomenon, but of low quality (e.g. by adding noise to the high quality data). If the Neural Network generalizes for the high quality data quite well, it will clearly produce poorer predictions with the low quality data, but the predictions are still reliable (although the prediction error will be larger), because the Neural Network had learned the essential features of the underlying system. This characteristic of generalization can even be detected from low quality data, because reliability means that the estimated prediction error can be trusted, even if it is high.

The third aspect of generalization concerns *model correctness*. Each trained Neural Network can be considered as a (formal) model representing the dynamics underlying the data. The generalization would obviously fail, if this formal model would be incorrect. In that case the predictions errors would show systematic deviations from the correct values. Accordingly this aspect of generalization is related to the correlation between errors and data.

To see how reliability and model correctness can be detected we consider the two types of generalization failures separately. First overadaptation is considered. This type of generalization failure is related to a particular trained network: As usual one splits the available data into *in-sample* and *out-of-sample* data, trains the Neural Netwark with the in-sample data and uses the out-of-sample data to measure the predictive performance. For the reliability of the predictions one has to look whether the prediction error can be trusted, or, expressed otherwise, whether the prediction error is stationary. This is usually only possible, if the data set is sufficiently large and this is often not the case. It is much simpler to test for model correctness. Here one has to check whether the prediction errors are uncorrelated to the data. It is easy to show that (linear) uncorrelatedness between errors and data is identical to an ideal correlation between predictions and data (besides an error in bias). Therefore the check for this aspect of generalization is identical to the usual check for model correctness by correlations (see e.g. Theil (1966)).

The second type of generalization failure, the inadequacy of network structure, leads to a different generalization check. The structure of a Neural Network is independent of the particular values of its various parameters. So, in contrast to the previous case, here not a particular (trained) Neural Network is considered, but a whole family of networks, all with the same structure. With respect to the network structure reliability means here, that the predictive performance of the

network is independent of the choice of the in-sample data. Clearly, for testing this, it has to be assumed that the training is optimal, i.e. that the particular Neural Network is neither under- nor overadapted. Assuming this, it is obvious how this aspect of generalization can be checked: Not only a single splitting of the data into in-sample and out-of-sample data has to be considered, but various different splittings. For each of these data splittings one trains the Neural Network with the in-sample data and determines the prediction errors for the associated out-of-sample data. For a good generalization with respect to the network structure the prediction errors should be independent of the particular data splitting. This can be checked, e.g., by considering the fluctuations of the mean prediction errors of the various sets of out-of-sample data. Only if these fluctuations are small, the prediction quality is independent from the particular splitting. An even simpler check would be to plot the prediction errors for overlapping out-of-sample data sets. Once more a small variation of the errors indicates a good ability to generalize with respect to network structure. This technique is well known in the time series literature under the names ”cross-validation” and ”v-leave-out” (see e.g.Weiss and Kulikowski (1990)).

These considerations show, that particular aspects of generalization can be checked, even if the quality of the predictions is low.

18.3 Automatic Termination of Training

As discussed above, to investigate whether a network structure is suited for a particular prediction problem, one has to perform crossvalidation studies. Unfortunately, using standard software with visual supervision of the training process, this is very laborious, because the training has to be repeated for many different training sets. In this situation it would be advantageous to automatize the training. Actually, the problem is not the training itself, but how to stop the training optimally to prevent under- and overadaptation. This problem is tackled by a stopping technique introduced by Weigend et al. (1991). But although this technique has been used in a number of studies (Weigend et al. 1991; Dodier 1994; Wan 1994), it seems, that nobody has tried to cast it into an algorithm

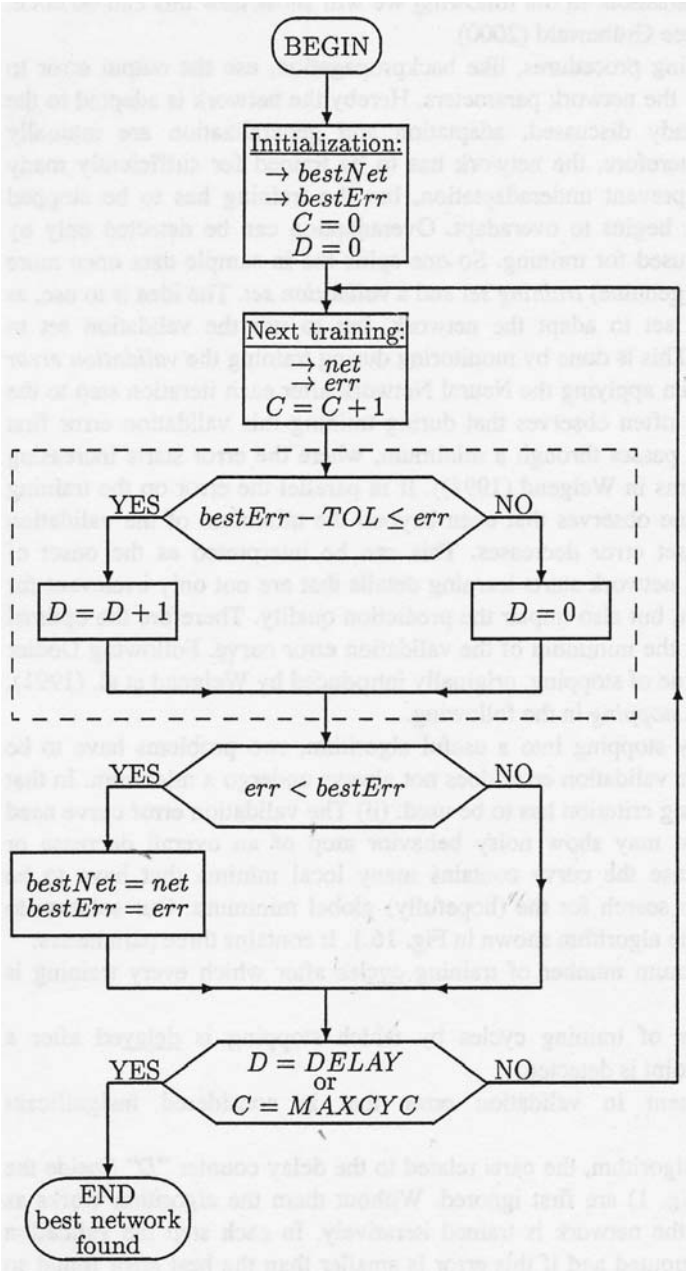


Fig. 18.1. Our early stopping algorithm.

suitable for automatization. In the following we will show how this can be done. For further details see Grünewald (2000).

Supervised training procedures, like backpropagation, use the output error to optimize iteratively the network parameters. Hereby the network is adapted to the data. But, as already discussed, adaptation and generalization are mutually excluding aims. Therefore, the network has to be trained for sufficiently many training cycles to prevent underadaptation, but the training has to be stopped before the network begins to overadapt. Overadaption can be detected only by data that were not used for training. So one splits the in-sample data once more into two parts: the (genuine) *training set* and a *validation set*. The idea is to use, as usual, the training set to adapt the network, but to use the validation set to terminate training. This is done by monitoring during training the *validation error* that is obtained when applying the Neural Network after each iteration step to the validation set. One often observes that during training this validation error first decreases and then passes through a minimum, where the error starts increasing (see e.g. the diagrams in Weigend (1994)). If in parallel the error on the training set is monitored, one observes that even beyond the minimum of the validation error the training set error decreases. This can be interpreted as the onset of overadaptation: the network starts learning details that are not only irrelevant for predicting new data, but also impair the prediction quality. Therefore the optimal stopping point is at the minimum of the validation error curve. Following Dodier (1994), this technique of stopping, originally introduced by Weigend et al. (1991), will be called *early stopping* in the following.

To convert early stopping into a useful algorithm, two problems have to be surmounted: (i) The validation error does not always undergo a minimum. In that case another stopping criterion has to be used. (ii) The validation error curve need not be smooth, but may show noisy behavior atop of an overall decrease or increase. In that case the curve contains many local minima that have to be escaped in order to search for the (hopefully) global minimum. Our solution to these problems is the algorithm shown in Fig. 18.1. It contains three parameters:

MAXCYC Maximum number of training cycles after which every training is stopped.

DELAY Number of training cycles by which stopping is delayed after a possible stopping point is detected.

TOL Improvement in validation error that is considered insignificant (”tolerance”).

To explain the algorithm, the parts related to the delay counter ”*D*” (inside the dashed frame in Fig. 18.1) are first ignored. Without them the algorithm works as follows: As usual the network is trained iteratively. In each step the validation error (”*err*”) is computed and if this error is smaller than the best error found so far (”*bestErr*”), the parameters of that – so far – best network and best error are saved (”*bestNet* = *net*”, ”*bestErr* = *err*”). The whole process is stopped, when the number of iteration cycles (”*C*”) reaches the maximum number of iteration cycles (”*MAXCYC*”). So far this algorithm describes simply a search for a best network that is stopped after a fixed number of training cycles and it is completely

irrelevant whether the validation error has a minimum, is ever increasing or shows a noisy behaviour with many minima. Insofar it has nothing to do with early stopping. Nevertheless, if *MAXCYC* is set to a very large value so that one can be sure that the global minimum of the validation error is reached (if it can be reached), this algorithm gives the same result as early stopping. But numerically it is not desireable to train the network in every case by the maximum number of cycles. Instead, one would like to stop the training already when the optimal network has been found. This cannot be fully achieved, but at least partially by including the so far ignored parts into the algorithm.

To explain their function, we first suppose *TOL* to be zero. When starting the algorithm the delay counter "*D*" is set to zero. If during training the validation error decreases ("*bestErr > err*") the delay counter stays zero ("*D=0*"). But if the error gets worse, i.e. if the validation error curve undergoes a minimum, the delay counter starts counting ("*D=D+1*"). If the validation error continues to increase, the delay counter will reach its maximum value *DELAY* and the training is stopped. So this is not really early stopping, but a delayed early stopping. The reason to perform some additional training cycles is to check whether the detected minimum is only a small local minimum or can be considered a global minimum. To understand how the algorithm detects this situation, assume that the delay counter already started counting, i.e. the network error already increased for *D* training cycles, and that in the present training cycle a validation error is found that is smaller than all previous validation errors ("*err < bestErr*"). Fig. 18.1 shows that in this case the delay counter is once more set to zero ("*D=0*"). Thereby the delay counter has to start once more from the beginning and the training will last at least *DELAY* additional training cycles (if not *MAXCYC* is reached before). Thereby the local minimum has been escaped. This means that for *TOL* equal to zero, the training stops only, if a detected minimum remains the absolute minimum for the next *DELAY* training cycles. If not, the next minimum will be checked. In this way training escapes, as desired, local minima.

Nevertheless, there is a problem: What happens, if the validation error is ever decreasing, or, decreasing on the average with local minima separated not farther than *DELAY* training cycles apart? In that case *D* would always be reset to zero, before *DELAY* has been reached. Accordingly, the algorithm continues training for the full *MAXCYC* training cycles. Such a behaviour is not always desireable, namely, when the error improves only insignificantly, so that even by longer training a substantially better network cannot be expected. To stop training in such cases before *MAXCYC* has been reached, the additional parameter *T OL* has been introduced. Once more the situation of a monotoneously decreasing validation error is considered, but now with *TOL* larger than zero. In this case the delay counter is reset to zero only if the validation error improves by an amont larger than *TOL*, i.e. in the case "*err < bestErr - TOL*". Therefore the training is stopped, if for *DELAY* training cycles the validation error improves less than *TOL*. Accordingly, as intended, a too long training is prevented, if no significant improvement can be expected.

It is clear that the choice of the three parameters *DELAY*, *MAXCYC* and *TOL* is decisive for a proper work of our early stopping algorithm. Therefore one has to perform some extra training runs to identify viable values for them. They should be chosen such that the training is stopped either because a ”global” minimum is detected, or because the network improvement gets insignificant but not because *MAXCYC* is reached; this parameter should guarantee only an ”emergency stop”.

18.4 Case Study: Zooplankton Prediction

In this section we apply crossvalidation and early stopping to an environmental data set from Greve (1988). The data document the zooplankton development from 1975 to 1994 at a position close to Helgoland island in the German North Sea. Every second or third workday the plankton was fished with a net of mesh size 150 µm. Simultaneously several other measurements were made: temperature, salinity etc. and in particular the water flow through the net. Each plankton organism from the catch was then visually inspected under a microscope, identified and counted. By the known water flow the number of individuals could then be converted into the density of organisms (individuals per cubic meter), called *abundance* in the following. The result is a large data set with the abundances of 45 groups of zooplankton organisms (”taxa”) at more than 3200 points of time. In addition we have data for two phytoplankton groups (diatoms and flagellates) from separate catches at the same position (measured as carbon mass per cubic meter) and data for seven physical parameters (water temperature, salinity, phosphate concentration, etc.). Unfortunately the catches were taken irregularly (all two or three days). To apply time series prediction methods one has to make the data equidistant. To this end we averaged all data of approximately one week (actually we averaged over 365.25/52=7.01923 days to account for intercalary days) so that the final data set has 52 data points for each of the twenty years.

The main problem with these data is that they are taken from a single point in a floating environment. Therefore the plankton organisms from two successive catches may not belong to the same population so that even on the level of populations there may be no deterministic relationship between two data points. Moreover it is known that plankton often comes in patches so that also the representativity of these random sample plankton measurements is questionable. And indeed, our extensive prediction studies of this data set indicate that short time predictions of the abundance are not possible. Nevertheless, if one restricts oneself to the prediction of only the order of magnitude of the abundances, a moderate prediction quality is achievable. For a given abundance x we define this *magnitude* m by

$$m = log_{10}(x+1) \qquad (18.1).$$

(The addition of "*1*" in this equation assures that abundance zero is mapped to magnitude zero; for large abundance the magnitude differs only negligibly from its decadic logarithm.) Accordingly, in the following we will consider only predictions of the magnitude of abundances and not of the abundances itself. Technically this was achieved by transforming all plankton data from abundances to magnitudes before averaging. This has also been done with the phytoplankton data.

It is generally expected that there is a complex network of interactions between the several types of plankton organisms ("food web"). Accordingly we tried to perform predictions by using simultaneously the time series of several taxa as input of the Neural Networks. In view of the large amount of possible combinations of taxa such studies could only be performed by massive automatization. Our aim here is not to present the results from this study (see Grünewald 2000). Instead we want to demonstrate the application of crossvalidation and early stopping to a particular prediction problem. The example we consider is the prediction of Barnacle larvae (*Cirripedia nauplius*) with a 16x5x2x1 feedforward Neural Network. As input we use the data from the last eight weeks of *Cirripedia nauplius* itself and also the last eight weeks of our diatom data. This combination was chosen because *Cirripedia nauplius* at least partially preys on diatoms. As learning algorithm we used "Resilient Backpropagation" (Riedmiller and Braun 1993), which is usually faster than the classical backpropagation algorithm.

From the 20 years of data we used the first 16 years as in-sample data and the last four years as out-of-sample data. For training the in-sample data were partititioned in 30 different ways into training and validation data (see Fig. 18.2); we used 12 years for training and 4 years for validation. The parameters used for the early stopping algorithm were *MAXCYC* = 1000, *MAXDEL* = 100 and *TOL* = 10^{-4}. Fig. 18.3 shows the predictions for the out-of-sample data obtained by the Neural Network that was trained with the data splitting no. 16 of Fig. 18.2. The predictions reproduce quite well the data. This is substantiated by a value of 0.91 for the correlation between the predictions and the data. Accordingly, the error, also shown in Fig. 18.3, is on the average significantly smaller than the data, although not small. At wintertime, where the error is of the same order as the data, better prediction results cannot be expected, because there are often less than 15 individuals per m^3 present, so that already the statistical uncertainty from sampling is of the order of the abundances. Overall, this prediction looks quite well.

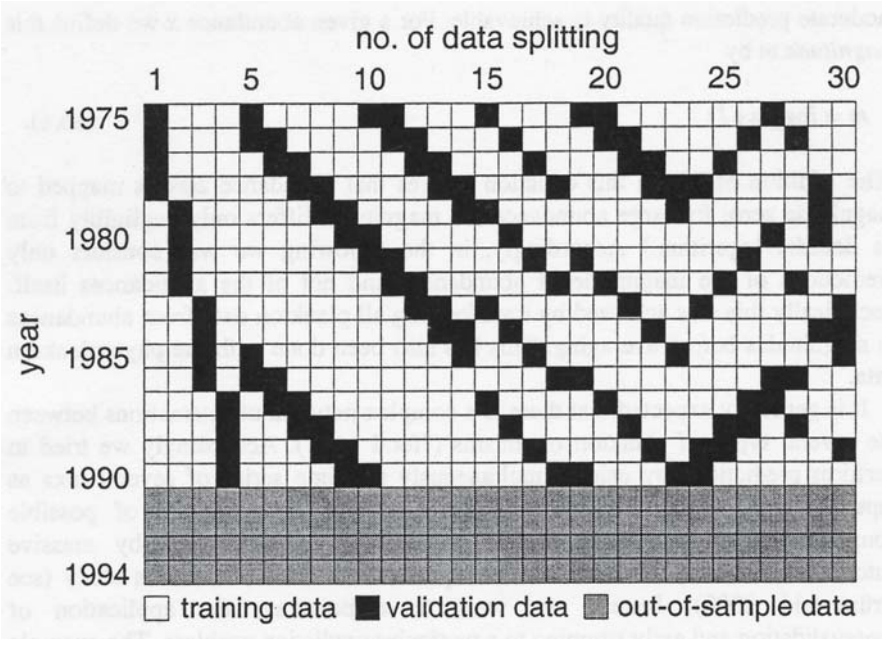


Fig. 18.2. Data splitting used for crossvalidation.

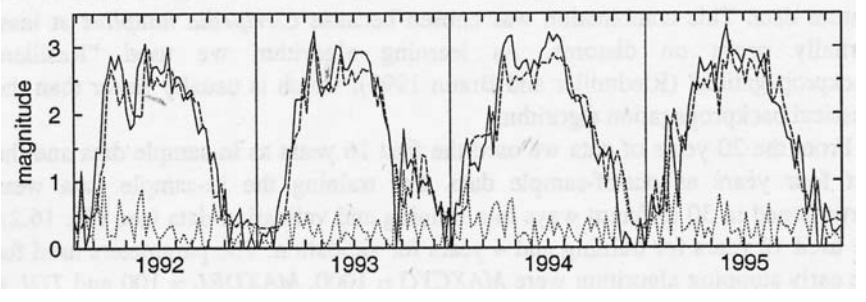


Fig. 18.3. Prediction of the magnitude of *Cirripedia nauplius* with a 16x5x2x1 Neural Network. Input are the data from the last eight weeks of both *Cirripedia nauplius* and *diatoms*. Full line: measurements; dashed line: predictions; dotted line: prediction error. For training the data splitting no. 16 of Fig. 18.2 has been used.

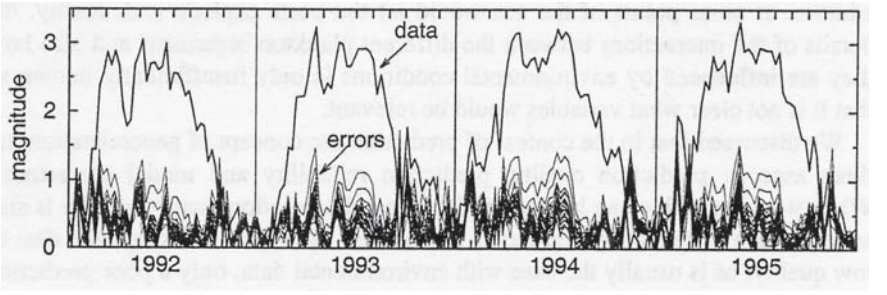


Fig. 18.4. Crossvalidation results for the prediction of *Cirripedia nauplius*. Shown are the data and the absolute errors of predictions by 30 Neural Networks. The structure of the networks was identical, but they were trained by different data splittings, namely those indicated in Fig. 18.2. The network structure was the same as in Fig. 18.3.

Nevertheless, a more critical look at the predictions changes this impression. A first indication that the generalization may not work well is the value 0.46 of the correlation between the data and the prediction error. This value is still significant so that the predictions cannot be considered to be independent of the data so that the modelling of the data by the Neural Network is only partially correct. And this is not the result of an insufficient training: the training was broken off by the early stopping algorithm at training cycle 364, because the last 100 cycles the validation error was increasing. The picture gets even worse, if one looks at the full crossvalidation results shown in Fig. 18.4. Obviously there are other splittings of the training data, for which the error is about 50% of the data maximum, so that the quality of the predictions depends strongly on the choice of the training data and the predictions are not reliable. Accordingly, one has to conclude that for this Neural Network generalization fails.

18.5 Conclusions

In this article we discussed the application of time series prediction by Neural Networks to environmental data sets. Such data are typically of low quality as compared to laboratory data for various reasons: boundary conditions for the phenomenon to be studied are not controllable, taking clean data is much too expensive or the phenomenon is so complex that the relevant variables that should be measured are not known. The consequence is: one usually has to work with the data one can get, and not the data one would like to have. The zooplankton forecasts considered in the previous section illustrate this situation: the data are from a completely uncontrollable environment, taking data more frequently or in addition at other points of the sea would let the costs explode and, finally, the

details of the interactions between the different plankton organisms and also how they are influenced by environmental conditions is only insufficiently known so that it is not clear what variables would be relevant.

We discussed that in the context of prediction the concept of generalization has three aspects: prediction quality, prediction reliability and model correctness. When working with clean laboratory data a good prediction quality usually is also an indication of reliability and model correctness. But when working with data of low quality, as is usually the case with environmental data, only a poor prediction quality can be expected so that the reliability and model correctness have to be considered independently of the prediction quality. Here crossvalidation techniques can be employed. Unfortunately, this is extremely laborous because the same Neural Network has to be trained for many different data splittings so that an automatization would be helpful. We showed that a major problem of automatization is to stop the training process automatically, such that the networks get neither under- nor overadapted. As a solution we proposed our early stopping algorithm and showed by an example how it can be applied in practice.

Actually, our early stopping algorithm is only a first step in the direction of automatized prediction studies. For large data sets with many variables one would need Neural Network tools that allow for complex definitions of large sequences of prediction experiments with networks of different structure, the automatic execution of these experiments as well as their automatic documentation and analysis. In the next years the various environmental monitoring programs with their automatic data aquisition will yield an ever increasing flood of data. By employing times series prediction methods, like those based on Neural Networks, these data could in principle be used for environmental management, e.g. to detect and anticipate significant ecosystem changes. But this will only be possible if appropriate Neural Network tools are available. And these have still to be developed.

Acknowledgements

We gratefully acknowledge the provision of the zooplankton data by W. Greve, Forschungsinstitut Senckenberg, Hamburg. This work was partially financed by the Bundesministerium für Bildung und Wissenschaft (BMBF) under the grant 03F0181B.

References

Armstrong JS (Ed.) (2000) Principles of Forecasting. Kluwer Academic, Boston

Dodier R (1994) Increase of apparent complexity is due to decrease of training set error. In: M.C. Mozer, P. Smolensky, D.S. Touretzky, J.L. Elman and A.S. Weigend (Editors),

Proceedings of the 1993 Connectionist Models Summer School, Erlbaum, Hillsdale, pp. 343-350

Gershenfeld NA, Weigend AS (1994) The future of time series: Learning and Understanding. In: A.S. Weigend and N.A. Gershenfeld (Editors), Time Series Prediction: Forecasting the Future and Understanding the Past, Addison-Wesley, Reading, pp. 1-70

Grünewald A (2000) Untersuchungen zur Prognostik mit Neuronalen Netzen. Diploma thesis, University of Hamburg

Greve W, Reiners F (1988) Plankton time-space dynamics in the German Bight. Oecologia 77: 487-496

Hertz J, Krogh A, Palmer RG (1991) Introduction to the Theory of Neural Computation. Addison-Wesley, Redwood City

Press WH, Flannery BF, Teukolsky SA, Vetterling WT (1986) Numerical Recipes. Cambridge University Press, Cambridge

Reick CH, Page B (2000) Time series prediction by multivariate next neighbor methods with application to zooplankton forecasts. Mathematics and Computers in Simulation, 52: 289-310

Riedmiller M, Braun H (1993) A direct adaptive method for faster backpropagation learning – The RPROP algorithm. In: H. Ruspini (Editor), Proceedings of the IEEE International Conference of Neural Networks (ICNN 93), San Francisco, pp. 586-591

Rost B, Sander C (1993) Prediction of protein secondary structure at better than 70% accuracy. J. Mol. Biology, 232: 584-599

Theil H (1966) Applied Economic Forecasting, North Holland, Amsterdam

Wan EA (1994) Time series prediction by using a connectionist network with internal delay lines. In: A.S. Weigend and N.A. Gershenfeld (Editors), Time Series Prediction: Forecasting the Future and Understanding the Past, Addison-Wesley, Reading, pp. 195-218

Weigend AS (1994) On overfitting and the effective number of hidden units. In: M.C. Mozer, P. Smolensky, D.S. Touretzky, J.L. Elman and A.S. Weigend (Editors), Proceedings of the 1993 Connectionist Models Summer School, pp. 335-342

Weigend AS, Rumelhart DE, Hubermann BA (1991) Backpropagation, weight elimination and time series prediction. In: D.S. Touretzky (Editor), Connectionist Models, Proceedings of the 1990 Summer School, Morgan Kaufmann, San Mateo, pp. 105-123

Weiss SM, Kulikowski CA (1990) Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning and Expert Systems, Morgan Kaufmann, San Mateo

Zell A (1994) Simulation Neuronaler Netze. Addison-Wesley, Bonn

Chapter 19

Classification of Fish Stock-Recruitment Relationships in Different Environmental Regimes by Fuzzy Logic with Bootstrap Re-sampling Approach

D.G. Chen

19.1 Introduction

The analysis of stock-recruitment (SR) relationships is a basic step in developing and evaluating fishery policies, such as establishing optimal escapement goals for salmon or optimal size of spawning stocks at maximum sustainable yield (MSY). Traditional SR analyses assume that there is a functional relationship between the size of the stock spawning biomass and the biomass of fish that recruit in the future. Numerous models have been discussed for this functional relationship. A comprehensive summary can be found from Ricker (1975), Hilborn and Walters (1992) and Quinn and Deriso (1999).

In the search for better tools for fish stock assessment, there has recently been a growing interest in the use of machine learning models (such as neural network models, fuzzy logic models and genetic algorithms) for research and management of natural resources (Lek et al. 1995; Mackinson et al. 1999 and Tang et al. 2000). It has been demonstrated that these models offer substantial advantages over traditional SR methods in model fit and forecast (Saila 1996; Chen and Ware 1999 and Chen et. al. 2000).

In this paper, the utility of fuzzy logic model with a hybrid global learning algorithm is explored to classify the SR relationships under different regimes for environmental and fishery management interventions. A bootstrap re-sampling scheme is also proposed to address the lack of uncertainty estimation in the machine-learning methods. The scheme produces a sampling probability distribution for the SR parameters related to fishery management policies so that the associated uncertainty measures (such as, variance, standard error, or confidence interval) can be obtained. Two SR applications: 1) southeast Alaska (SEAK), USA, pink salmon, and 2) west coast Vancouver Island (WCVI), BC, Canada, herring, are examined to demonstrate the advantages of this new model to

the traditional approaches. In both examples, the annual mean sea-surface temperature (SST) is incorporated as an environmental intervention.

19.2 Fuzzy Stock-Recruitment Model

19.2.1 Traditional Stock-Recruitment Model

SR analysis begins with the assumption of a functional relationship, denoted by F(•), between spawners and recruitment:

$$R_t = F(S_t, \theta) \tag{19.1}$$

where R_t and S_t are the corresponding recruits and spawners at brood year t (t=1, ⋯, n), θ is a vector of parameters associated with this relationship and usually θ is associated with the fishery management policy. The Ricker model (1975) (hereafter referred to as Ricker-SR) is the most commonly used form in the fisheries literature:

$$R_t = S_t \, exp(a - b \, S_t) \, exp(\varepsilon_t) \tag{19.2}$$

where a is the parameter measuring fish stock reproductive performance at low stock size with exp(a) the maximum recruits per spawner and b is the parameter representing density-dependence in juvenile survival rate; and ε_t is a normally distributed "process" error with mean 0 and standard deviation σ. This model can be linearized as:

$$y_t = \log\left(\frac{R_t}{S_t}\right) = a - b \, S_t + \varepsilon_t \, . \tag{19.3}$$

The parameters a and b can be estimated by simple least-squares regression. Having estimates of a and b, fishery management parameters, such as the optimal stock size at maximum sustainable yield (MSY), S_{MSY}, and harvest rate, μ_{MSY}, can be calculated for species that die after spawning based on the formulations from Hilborn (1985), Hilborn and Walters (1992) and Quinn and Deriso (1999):

$$S_{MSY} = \frac{a(0.5 - 0.07a)}{b} \qquad \text{and} \quad \mu_{MSY} = a(0.5 - 0.07a). \tag{19.4}$$

There is an increasing awareness that changes in environmental and fishery conditions can impact SR relationships. It is now known that fishery SR relationships have been masked by environmental and fishery management interventions. Fish recruitment is not only related to numbers of spawners in the parental generation, but is also influenced by environmental factors (e.g. sea surface water temperature and salinities) controlling natural survival and fisheries (Koslow et al. 1986; Ware and McFarlane 1995; Ware 1996; Ryall et al. 1999; Chen and Ware 1999; Chen et al. 2000). Therefore, the means to incorporate these interventions into SR analysis and to classify the SR relationships for different environmental regimes are becoming increasingly important. The procedures to incorporate these interventions into the SR analyses are summerized in Chen and Irvine (2001). This paper will be concentrated on the classification of the SR relationship into different regimes. The commonly used approach in the classification is to subset the SR relationships with various types of average for the classification of the intervention (such as by SST, salinity) (hereafter referred to as crisp classification). For example, Ware (1996) utilized the long-term time series average of the environmental factor (e.g. SST) to categorize the SR into two different regimes: "Warm Years" and "Cool Years". Schweigert and Noakes (1990) briefly discussed discriminant function models for the "Poor", "Average" and "Good" recruitment groups, which is obtained by ranking the recruitment from the lowest to the highest and assigning first one-third SR data to the "Poor" subgroup, the middle one-third to the "Average" subgroup and the last one-third to the "Good" subgroup. The very same classification was used in Hyatt at al. (1994) in the forecast and assessment for Barkley Sound sockeye from British Columbia, Canada. Four fundamental problems originated from these crisp approaches. Firstly and most importantly, the data observed for the environmental variable might be just a short time series of the real world representations and the crisp classifications based on the observed data have high possibility for misclassification. Secondly, the crisp approach oversimplifies the natural characteristics of the environmental interventions and it is easy to misclassify those years close to the thresholds. Using the SST data from the west coast of Vancouver Island as a simple example, since the long-term time series average for SST is 10.45°C (Fig. 19.1), then the years of 1982 and 1991 with SST of 10.40°C and 10.42°C, respectively, would be classified as "Cool Years" and the years of 1961, 1962 and 1977 with SST of 10.47 °C, 10.50°C and 10.54°C would be classified as "Warm Years". The misclassification could even be serious for the years with SST of 10.45°C since it would be difficult to classify them into either category. Thirdly, this approach embedded the disadvantage that the information from SST is ignored in the process of fitting the data using equation (19.3). And finally, with this crisp classification, the SR data from the "Warm Years" are not used in fitting the SR model to the data from "Cool Years" and verse visa.

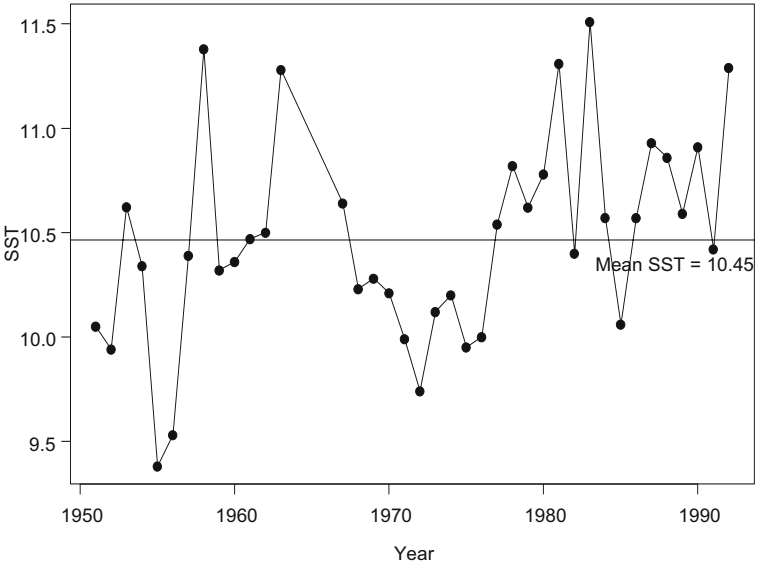


Figure 19.1. Time series of the annual mean sea surface temperature (°C) at Amphitrite Point, west coast of Vancouver Island. The horizontal line is the long-term time series average. Note year 1977 is the year for the transition to the current warm climate regime.

In general, most of these environmental factors are intrinsic fuzzy terms and there is no crisp and clear break point for the classification. Therefore a fuzzy logic approach should lead to an improved SR analysis.

19.2.2 Fuzzy Stock-Recruitment Model

A fuzzy logic model is also known as a fuzzy inference system or fuzzy-rule-based system. Basically, any fuzzy logic model consists of three parts, which are the fuzzy membership functions, fuzzy decision rules, and the fuzzy reasoning. Several types of fuzzy reasoning have been developed in the literature (Bandemer and Gottwald 1995; Lee 1990). Following the traditional SR model (19.3), a fuzzy logic SR model (hereafter referred as Fuzzy-SR) is proposed in this paper to model and classify the fish SR relationship. Without loss of generality, the description of this Fuzzy-SR model is restricted to only two environmental regimes, i.e. such as “Cool” and “Warm”. The extension to any number of regimes can be easily made with corresponding modifications to the fuzzy membership functions, fuzzy decision rules, and the fuzzy reasoning.

19.2.2.1 Fuzzy Membership Function (FMF)

Corresponding to the traditional treatment for the environmental variables in the SR analyses, only SST is used as fuzzy input and the stock spawner biomass (S) and recruitment (R) are kept as crisp variables.

The logistic membership function for the input variable SST is used for the fuzzy partition as "Cool" and "Warm". Specifically the symmetrical membership functions for "Warm" and "Cool" are defined as:

$$FMF_{Warm}(SST,\alpha,\beta)=\frac{1}{1+\exp[-\beta(SST-\alpha)]}; \tag{19.5}$$

$$FMF_{Cool}(SST,\alpha,\beta)=\frac{1}{1+\exp[\beta(SST-\alpha)]} \tag{19.6}$$

where parameter α is used to describe the mean SST and parameter β is used to describe the slope of the membership function (Fig. 19.2). It can be easily shown that

$FMF_{Warm}(\mathrm{SST}, \alpha, \beta) + FMF_{Cool}(\mathrm{SST}, \alpha, \beta) = 1$. And if $\beta \to \infty$,
$FMF_{Warm}(\mathrm{SST}, \alpha, \beta) \to I(SST-\alpha)$ and $FMF_{Cool}(\mathrm{SST}, \alpha, \beta) \to I(\alpha - SST)$ where I(x) is the indicate function defined as $I(x) = 0$ if $x<0$ and $I(x) = 1$ if $x \geq 0$.

It is worth noting that for this case, the fuzzy implications of equations (19.5) and (19.6) return to the crisp classification, which is that if SST is lower than the long-term time-series average (α), SST is "Cool", otherwise, it is "Warm" (the dashed line in Fig. 19.2). Therefore, the Fuzzy-SR model is an extension of the traditional SR model (Ware 1996; Schweigert and Noakes 1990; Hyatt at al. 1994) (hereafter referred as Crisp-SR).

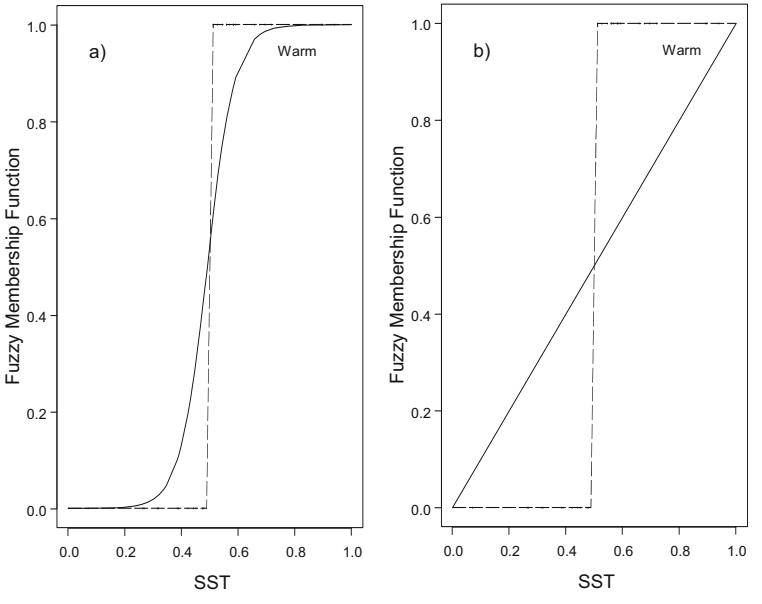


Figure 19.2. a) is the fuzzy membership function of "Warm" for SST (WCVI herring stock) after the hybrid optimization with resultant estimates: $\alpha = 0.49$ and $\beta = 20.82$; b) is the fuzzy membership function of "Warm" for SST (SEAK pink salmon stock). The dashed line in both plots illustrates the crisp classification of "Warm" based on the long-term time series average (i.e. "Cool Years" if SST is less than the average, otherwise, "Warm Years").

19.2.2.2 Fuzzy Rules

There are two fuzzy rules that are totally determined by the choice of the fuzzy membership functions defined in Section 19.2.2.1. In general form, each fuzzy rule is written as:

$$\text{Rule 1: If } SST_t \text{ is "Cool", then } y_t = a_1 - b_1 S_t; \tag{19.7}$$

$$\text{Rule 2: If } SST_t \text{ is "Warm", then } y_t = a_2 - b_2 S_t; \tag{19.8}$$

where a_1, a_2, b_1 and b_2 are fuzzy parameters to be estimated. In fact a_1 and a_2 are the parameters corresponding to the "Cool" and "Warm" regimes to measure fish stock reproductive performance at low stock size. Furthermore, $\exp(a_1)$ is the maximum recruits per spawner for the "Cool" regime and $\exp(a_2)$ the maximum

recruits per spawner for the "Warm" regime. The parameter b_1 and b_2 represent density-dependence in juvenile survival rate in "Cool" and "Warm" regimes,

respectively. The function $y_t = \log\left(\frac{R_t}{S_t}\right)$, t =1 to n, is the log-transformed stock

productivity. With the rules defined in (19.7) and (19.8), the "consequent" parts of the two fuzzy rules are defined by the non-fuzzy equations of the stock spawner biomass, which is similar to the definition given by Takagi and Sugeno (1983).

19.2.2.3 Fuzzy Reasoning

With the above "implications" Rule i (i = 1 to 2) and for any observed SST_t and corresponding S_t (the fish spawner biomass), the model value of y is then inferred from the following steps:

Step 1: The firing level (weight) for Rule i is computed by:

Rule 1: $1\text{-}w_t = FMF_{Cool}(SST_t, \alpha, \beta)$

Rule 2: $w_t = FMF_{Warm}(SST_t, \alpha, \beta)$;

Step 2: For each Rule i, $\hat{y}_{ti}$ is calculated by the function defined in (19.7) and (19.8):

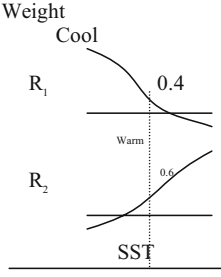
$$\hat{y}_{ti} = a_i - b_i S_t$$

Step 3: The final output of the Fuzzy-SR system, $\hat{y}_t$, that is inferred from the two rules is computed by the weighted average defuzzification method as

$$\hat{y}_t = (1 - w_t)\,\hat{y}_{t1} + w_t\,\hat{y}_{t2}. \qquad (19.9)$$

This process is summarized in Table 19.1. With the defined Fuzzy-SR model, the parameters from the FMF (e.g. α, β) as well as fuzzy parameters (a_1, a_2, b_1 and b_2) can be estimated by any optimization procedures.

Table 19.1. Fuzzy reasoning process. Column "Implication Premise" describes the fuzzy membership function for SST under two fuzzy rules; column "Consequence" is the value calculated from each consequence for the inputs and corresponding parameters, and column "Weight" is calculated from the fuzzy memberships from the input.

Implication Premise	Consequence	Weight
	$y_1 = a_1 - b_1 S$	1- w = 0.4
	$y_2 = a_2 - b_2 S$	w = 0.6

19.3
Hybrid Optimal Learning and Bootstrap Re-sampling Algorithms

The objective for the learning algorithm is to optimize some error measures (or energy functions), which is mostly the sum of squares of errors (SSE):

$$E(\alpha,\ \beta;\ a_1,\ a_2,\ b_1,\ b_2\ ) = \sum_{t=1}^{n}\left(y_t - \hat{y}_t\right)^2 = \sum_{t=1}^{n}\left[\, y_t - (1-w_t)(a_1 - b_1 S_t) - w_t\,(a_2 - b_2 S_t)\,\right]^2 , \tag{19.10}$$

where y_t is the observed fish recruitment biomass and $\hat{y}_t$ is the Fuzzy-SR modelled value from (19.9), which is a function of unknown parameters α, β (from the FMF of SST) and a_1, a_2, b_1, b_2 (fuzzy parameters). The estimation of these parameters is obtained from minimizing (19.10), which is equivalent to the classical non-linear least-squares estimation (LSE). However, it is well known that it is difficult to find the global optimal set of parameters, especially when there are a large number of local minima. In such instances, conventional mathematical search algorithms are likely to converge on some local minima, instead of the global minima. In the search for a better optimal algorithm, Chen et al. (2000) discussed the application of genetic search algorithms to SR fitting and forecasting. Although genetic search algorithms are global optimization algorithms, it has been found that the algorithms do require a large amount of computer time to find the global optima. In a situation requiring bootstrapping which includes re-sampling the SR data for a large number of times, the genetic algorithms do not seem practical. In this paper, a hybrid optimal learning algorithm which combines the gradient descent and linear least-squares estimation (LSE) is adopted to search for the global optima and also for the bootstrap re-sampling procedure.

19.3.1
Hybrid Optimal Learning Algorithms

The gradient method is the basic learning algorithm for any optimization (Press et al. 1988). To implement the gradient descent, the *error rate,* $\frac{\partial E}{\partial P}$, needs to be calculated for each parameter ($P = \alpha, \beta, a_1, a_2, b_1, b_2$), which can be easily obtained from equation (19.10). Then, the optimal parameter estimate can be learned as:

$$P^{(k+1)} = P^{(k)} - \eta \frac{\partial E}{\partial P^{(k)}}, \tag{19.11}$$

where $P^{(k)}$ indicates the kth updates for parameter P and η is a learning rate to vary the speed of convergence. However, the gradient descent method is notorious for its slowness to converge and tendency to be trapped in local minima if there is no prior information for the parameters.

Because of the linearity of the fuzzy rules in (19.7) and (19.8), the gradient descent learning algorithm can be combined with the linear LSE to calculate the global optima. It can be seen from equation (19.10) that for fixed parameters α and β, the minimization of (19.10) to obtain the parameter estimates for a_1, a_2, b_1 and b_2 is equivalent to the linear LSE, which is:

$$Y = XB, \tag{19.12}$$

where $Y = (y_1, \cdots, y_n)'$ is a $n \times 1$ vector of observed fish stock productivity indicate defined as $y_t = \log\left(\frac{R_t}{S_t}\right)$, $B = (a_1, b_1, a_2, b_2)'$, is an 4×1 parameter vector, and

$$X = \begin{pmatrix} 1-w_1, -(1-w_1)S_1, & w_1, & -w_1 S_1 \\ \vdots & \vdots & \vdots & \vdots \\ 1-w_n, -(1-w_n)S_n, & w_n, & -w_n S_n \end{pmatrix}, \text{ is a } n \times 4 \text{ matrix}$$

constituted by observed fish spawner biomass and known parameters α and β from FMF. The global optimal parameter estimate can be obtained from:

$$\hat{B} = (X'X)^{-1} X'Y \,, \tag{19.13}$$

under the assumption that $X'X$ is non-singular. Otherwise $\hat{B}$ in (19.13) becomes ill-defined. To deal with this problem, the sequential method (Goodwin and Sin 1984; Strobach 1990) is adopted here. Specifically this calculates iteratively the following sequential formulas:

$$B_{i+1} = B_i + V_{i+1}\, x_{i+1} (y_i - x_{i+1}^{'} B_i \,)$$

$$V_{i+1} = V_i - \frac{V_i\, x_{i+1}\, x_{i+1}^{'} V_i}{1 + x_{i+1}^{'} V_i\, x_{i+1}}, \;\; i = 0,1,\cdots,n-1 \tag{19.14}$$

where V_i is called the *covariance matrix* and $x_i^{'}$ is the *i*th row vector of matrix X. The initial conditions for the sequential equations (19.14) are $B_0 = 0$ and $V_0 = \gamma I$, where γ is a large positive number and I is the 4 × 4 identity matrix. The LSE for $\hat{B}$ is then equal to B_n in (19.14).

It is well known that the LSE for parameters: a_1, a_2, b_1 and b_2 are the global optima for (19.10) if the FMF parameters α and β are known in advance. However, according to the definition of FMF (19.5) and (19.6), α is the parameter to describe the long-term time series average and β is the parameter to describe the slope of the logistic curve. Therefore there is good prior knowledge to be used for the initial values and there is a high possibility to reach the global optima. Now the hybrid optimal algorithm to learn the global optima for the FMF and fuzzy parameters can be initialized to combine the steepest gradient descent method and the linear LSE as follows. Each iteration of this hybrid learning procedure consists of a forward pass and a backward pass. In the forward pass, the initial values for α and β are initialized and the observed data for (S_t, R_t, SST_t) are specified for a forward calculation to get X and Y in equation (19.12). Then the optimal estimates for fuzzy parameters: a_1, a_2, b_1 and b_2 can be attained by the sequential LSE in (19.14). After attaining fuzzy parameters: a_1, a_2, b_1 and b_2, the calculations keep going forward until the SSE in (19.10) is obtained. Then in the backward pass, the error rates for α and β propagate from the output end toward the input end, and the estimates for α and β are updated based on the steepest gradient descent (19.11).

The proposed hybrid learning algorithm is guaranteed to find the global optima for the fuzzy parameters (i.e. a_1, a_2, b_1 and b_2) and also the FMF parameters (i.e. α and β) if reasonable prior knowledge is available for the fuzzy membership

function, which is most often the case. In fact, this hybrid learning algorithm not only decreases the dimension of the search space in the gradient method, but it also substantially reduces the convergence time.

19.3.2 Bootstrap Re-sampling Procedure

For any reasonable and intelligent fisheries management implementation, it is wise and prudent not only to give a point estimate for the fishery policy parameters, but also to give a measure of the uncertainty for these management policy parameters. The most commonly adopted measures are the standard errors, confidence interval and even a probability distribution. Of course, if the probability distribution can be obtained, the associated standard error and confidence interval can be readily calculated. However, to my knowledge, this is not a common practice in most machine-learning methods. To address the lack of uncertainty estimation in the machine-learning models (the Fuzzy-SR model in this paper), a bootstrap re-sampling scheme is proposed to produce a sampling probability distribution for the stock parameters related to fishery management policies so that the associated variance (or standard error) and confidence interval can be obtained since bootstrap resampling is widely used to obtain such sampling distributions. Efron and Tibshirani (1993), Shao and Tu (1995), and Davision and Hinkley (1997) provide extensive theoretical backgrounds and plenty of examples.

In general, the available data (S_t, R_t, SST_t) to the Fuzzy-SR model can be treated as either deterministic or random variables. In the case of deterministic variables, the residuals $\varepsilon_t = y_t - \hat{y}_t$, are assumed to be identically independently distributed (*i.i.d.*) with a zero mean and a constant variance of σ^2. In the case of random variables, data (S_t, R_t, SST_t) are assumed to be *i.i.d.* with $E(\varepsilon_t|S_t, SST_t)=0$. The above assumptions correspond to two different types of settings for bootstrap resampling (Tibshirani 1994). One setting treats the inputs as fixed based on the deterministic variables (S_t, SST_t) with the model residuals $\varepsilon_t = y_t - \hat{y}_t$ as the sampling units, which are called *bootstrap residuals*. The other setting treats each data point (S_t, R_t, SST_t) as a sampling unit, which is commonly called *bootstrap pairing*. Since most SR data are intrinsically auto-correlated based on the spawner and recruitment interactions, then the *bootstrap residuals* would be more appropriate with the diagnostics of the residuals from the Fuzzy-SR model. This *bootstrap residuals* re-sampling strategy involves following steps:

Step 1: Construct the Fuzzy-SR model from Section (19.2.2) from the original SR data (S_t, R_t, SST_t; t=1 to n) and obtain the parameter estimates (i.e. a_1, a_2, b_1, b_2, α and β) from the hybrid optimal learning algorithm in Section (19.3.1);

Step 2: Calculate the residuals, $\varepsilon_t = y_t - \hat{y}_t$. Perform the residual diagnostics for independence and homogeneity. If the residuals are identically independently distributed (*i. i. d.*) with a zero mean and a constant variance of σ^2, then go to *Step* 3. Otherwise, go back to *Step* 1 with a proper transformation for the SR data;

Step 3: Randomly draw an *i. i. d.* sample $\{ \varepsilon^*_t \}^n_{t=1}$ with replacement from the residuals $\varepsilon_t = y_t - \hat{y}_t$ and construct the matrix X and Y* = $[y^*_1,\ldots,y^*_n]$ in equation (17.12) with $y^*_t = \hat{y}_t + \varepsilon^*_t$, t = 1,…,n;

Step 4: Using the procedure from (19.14) to get a new set of parameter vector B=(a_1, a_2, b_1, b_2) with the resampled data;

Step 5: Repeat Step 3 to Step 4 a large number of times, say, N, (with 1000 as a suggested number of repeats).

The above steps will yield a sample for the Fuzzy-SR parameter vector as $B_1, \cdots, B_N$. This sample can be used to construct a sampling distribution for the SR parameters: a_1, a_2, b_1 and b_2. The sampling distributions can then be obtained for the fishery management policy parameters, such as MSY spawner, S_{MSY} and MSY exploitation rate, u_{MSY} from equation (19.4).

19.4
Two Real Data Analyses

19.4.1
West Coast Vancouver Island Herring Stock

19.4.1.1
Data Prescription and Preliminary Analyses

It was found from a long-term research program of the west coast of Vancouver Island (WCVI), British Columbia herring stock that the SST (in year t-3) has profound impact on the biomass of 3-year old herring recruits (in year t) along with the biomass of spawners (i.e. parents) in the year in which the recruits were born (Ware 1991; Ware and McFarlane 1995; Chen and Ware 1999). Temperature is believed to be a proxy "signal" which reflects inter-annual variability in the relative biomass of larval and juvenile herring predators, and possibly some important components of the herring food supply. In general, cooler (warmer) temperatures tend to produce larger (smaller) recruitments 3-years later. This negative correlation between the SST and the biomass of WCVI herring recruits has been a consistent feature of the recruitment time series, since Tester (1948) discovered it about 50-years ago. Recent work indicates that the relationship between parent spawners and recruits is masked by the temperature effect (Ware 1996; Chen and Irvine 2001). The underlying dome-shaped SR relationship becomes apparent if the recruitment data are sorted into two groups: year-classes born in years of above average temperature, and year-classes born in

years of below average temperature (Fig. 19.3a). For each group there is a significant Ricker-like relationship between spawner and recruit biomass.

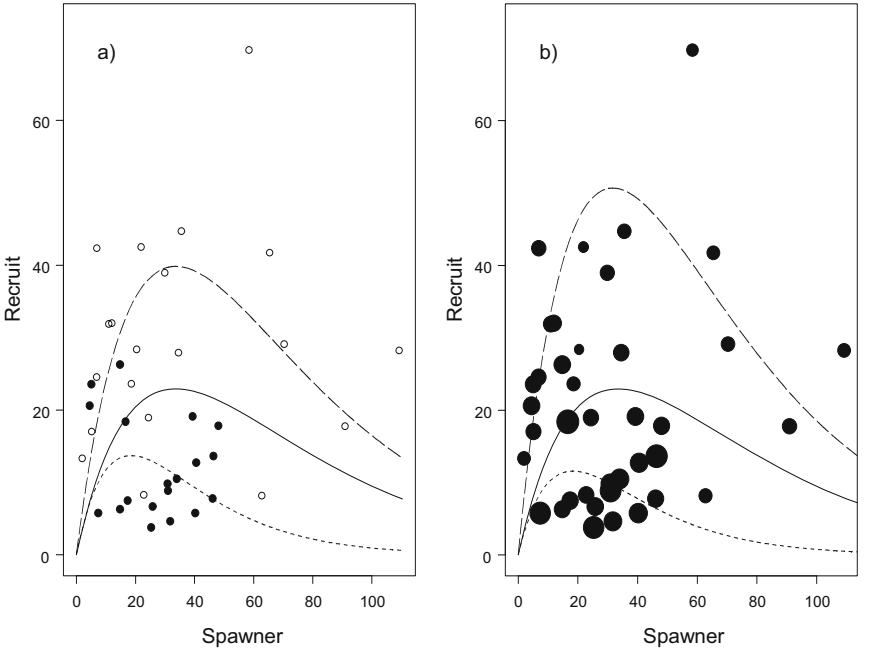


Figure 19.3. a) is for the Crisp-SR model. The bullets (•) are the SR data corresponding to the "Warm Years" and the circles (o) are the SR data corresponding to the "Cool Years". The lines from the top to the bottom are the fitted lines from Ricker model for the "Cool Years", all data combined and "Warm Years". b) is for the Fuzzy-SR model. The bullets (•) are the SR data with the radius in proportion to the magnitude of SST in that the higher the SST, the larger the radius. The lines in the top and the bottom are the Fuzzy-SR model fitted lines and the line in the middle is the fit from the simple Ricker model to all data combined.

19.4.1.2 Fuzzy-SR Model Analysis

To implement the Fuzzy-SR model, the input data SST for the fuzzy operation is re-scaled from 0 to 1. The fuzzy membership function is illustrated in Figure 19.2a corresponding to the re-scaled data. This FMF in Fig. 19.2a is also standardized to the range 0 to 1 with $FMF_{Warm}(0, \alpha, \beta)=0$ and with $FMF_{Warm}(1, \alpha, \beta)=1$. Following the Fuzzy-SR model described in Section (19.2.2), the fuzzy parameters (a_1, a_2, b_1 and b_2) and FMF parameters (α and β) are learned from the

procedures described in Section 19.3. In the optima learning process, the initial value for the centre-parameter α is chosen to be the long-term time series average and initial slope parameter β is chosen to be 200 to reflect the crisp classification. Figure 19.3b) illustrates the performance of the Fuzzy-SR model. The estimated parameters as well as the summary statistics are summarized in Table 19.2. For comparison, a Ricker SR model was also fit to the data.

Table 19.2. Summary of the model fits from the Ricker SR model (Ricker-SR), the SR model with crisp classification (Crisp-SR) and the fuzzy SR model in this paper (Fuzzy-SR). For Ricker-SR, there is only one set of *a* and *b*, which is placed under a_1 and b_1 in the table. The value inside the bracket is the estimated standard error, which is obtained from the simple linear regression for Ricker-SR and Crisp-SR. The standard error for Fuzzy-SR is the calculated standard error from the bootstrap sample. NA indicates that the value is not applicable. In the table RMSE is the rooted mean squares of errors, AIC is value from Akaike information criterion and r is the correlation coefficient.

	WCVI Herring			SEAK Pink		
	Ricker-SR	Crisp-SR	Fuzzy-SR	Ricker-SR	Crisp-SR	Fuzzy-SR
RMSE	16.18	12.13	10.98	5828.16	5004.98	4709.74
AIC	221.13	202.69	198.92	524.23	519.09	515.44
r	-0.12	0.58	0.68	0.43	0.64	0.68
a_1	0.62 (0.25)	1.17 (0.23)	1.47 (0.23)	1.05 (0.28)	0.57 (0.48)	0.52 (0.33)
b_1	2.97E-2 (6.2E-3)	2.97E-2 (4.90E-3)	3.16E-2 (4.90E-3)	5.52E-5 (7.89E-5)	2.58E-6 (1.52E-4)	1.23E-4 (1.07E-4)
a_2	NA	0.70 (0.37)	0.54 (0.29)	NA	1.49 (0.31)	1.72 (0.28)
b_2	NA	5.40E-2 (1.20E-2)	5.46E-2 (9.47E-3)	NA	1.03E-3 (8.15E-4)	6.82E-5 (7.59E-5)
♋	NA	NA	0.49	NA	NA	NA
♌	NA	NA	20.82	NA	NA	NA

A comparison of the results produced by the three different models is summarized in Table 19.2. The model comparison is based on three criteria. The first is the estimated root-mean-square-error (RMSE) in which the smaller the RMSE, the better the model fit. The second criterion is the well-known Akaike

information criterion (AIC) (Sakamoto et. al. 1986) to penalize MSE from the number of the model parameters. When using AIC criterion to compare the model fit, the smaller the AIC, the better the fit. The third criterion is the correlation coefficient (r). From all these three criteria, the Fuzzy-SR model produced the best fit to the original recruitment time series. The Crisp-SR model performed better than the Ricker-SR model, which is consistent with the results obtained by Ware (1996).

Table 19.3. Summary statistics of the bootstrap re-sampling distributions corresponding to two different environmental regimes: "Cool" and "Warm". The column "Model" represents the estimate from the Fuzzy-SR model, "Mean" is the mean for the 1000 bootstrap samples and "CI" is the 95% sample confidence interval obtained from 2.5% and 97.5% sample quantiles.

		WCVI Herring			SEAK Pink		
		Model	Mean	CI	Model	Mean	CI
Cool	a_1	1.47	1.46	(0.98, 1.92)	0.52	0.61	(0.103, 1.3)
	b_1	3.16E-02	3.13E-02	(2.28E-2, 4.07E-2)	1.23E-04	1.55E-04	(6.12E-6 ,3.98E-4)
	S_{MSY}	NA	NA	NA	1960	1808	(484, 3770)
	μ_{MSY}	NA	NA	NA	0.24	0.27	(0.05, 0.53)
Warm	a_2	0.54	0.59	(0.64,1.17)	1.72	1.84	(1.36, 2.49)
	b_2	5.46E-02	5.61E-02	(3.38E-2, 7.61E-2)	6.82E-05	1.03E-04	(5.27E-6 ,2.91E-4)
	S_{MSY}	NA	NA	NA	9573	8589	(2670, 24600)
	μ_{MSY}	NA	NA	NA	0.65	0.68	(0.55, 0.81)

19.4.1.3 Bootstrap Re-sampling Analysis

The residuals from the Fuzzy-SR model are diagnosed for the independence and homogeneity. The independence for the residuals can be checked by the time series autocorrelation function (Box et. al. 1994) and the homogeneity of residuals can be identified from the residual plot and also the Kolmogorov- Smirnov goodness-of-fit test. For this data, there is no violation for the assumptions. Therefore the *bootstrap residuals* procedure in Section (19.3.2) is legitimate to carry out for N = 1000 times. The bootstrap sampling distributions for the Fuzzy-SR parameters: a_1, a_2, b_1 and b_2 are illustrated in Fig. 19.4 (the first two rows in Fig. 19.4). These bootstrap samples can be readily used to obtain the uncertainty estimate, such as confidence intervals and standard errors (Table 19.3).

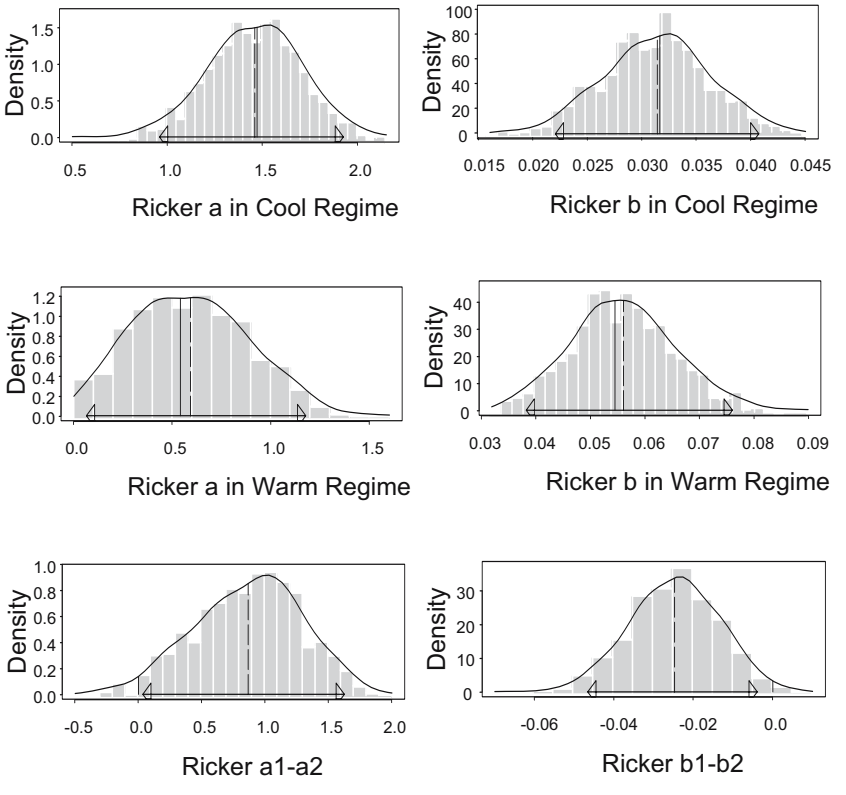


Figure 19.4. Bootstrapping sampling distributions for fuzzy parameters: a_1, b_1, a_2, b_2, a_1-a_2 *and* b_1- b_2 from 1000 bootstrap samples. In each plot, the line at the top of the histogram bars is the kernel density estimate of the probability density function. The horizontal line with open arrows in the end denotes the 95% sample confidence interval. The sample means from the bootstrap samples are marked as the dashed vertical lines. The vertical solid lines for the first two rows are the parameter estimates from the Fuzzy-SR model. The vertical solid lines for the last row are from zero to test whether the difference between the two parameters is statistically significant.

In Fig. 19.4, the 95% confidence intervals are marked by the horizontal lines with open arrows and the sample standard errors for the fuzzy parameters are also listed in Table 19.2. It can be seen from Table 19.3 and also Fig. 19.4 that all the model parameters are statistically significant.

Furthermore these bootstrap samples can be used to test the significance of the environmental impact on SR relationships from different environmental regimes.

This is carried out by testing the difference of the Ricker a and b between the "Warm" and "Cool" regimes. The last row in Fig. 19.4 is the distributions for a_1-a_2 and b_1-b_2 marked with the 95% confidence intervals, which are (0.034, 1.62) and (-0.045, -0.004), respectively. Since both intervals do not cover zero, then the differences for a_1-a_2 and b_1-b_2 are statistically significantly different. Specifically WCVI herring is more productive and less density-dependent in "Cool" regime than in "Warm" regime (a_1 is significantly larger than a_2, and b_1 is significantly less than b_2).

19.4.2 Southeast Alaska Pink Salmon

19.4.2.1 Data Description and Preliminary Analysis

Detailed data descriptions and preliminary analyses for Southeast Alaska (SEAK) pink salmon (*Oncorhynchus gorbuscha*) can be found from Quinn and Deriso (1999, p104–123). The SR time series is reproduced in Figure 17.5. To account for some of the unexplained variation in recruitment, Quinn and Deriso introduced an environmental factor: average annual sea surface temperature (SST) off Sitka, Alaska into the analysis. They found that a Ricker climatic SR model produced a statistically significant fit to the recruitment time series.

In order to determine the impact of different environmental regimes, these SR data are sorted into two subclasses: year- classes born in years of above average SST, and year-classes born in years of below average SST (Fig. 19.5a). The Crisp-SR approach is then fitted to these two data sets. It is found that there exist two different SR relationships with the stock productivity parameter for "Warm Years" 1.49 and "Cool Years" 0.57. Also the Crisp-SR model fits better than the Ricker-SR model, which is concluded from a decrease in the RMSE and an increase in the correlation coefficient (r) (Table 19.2).

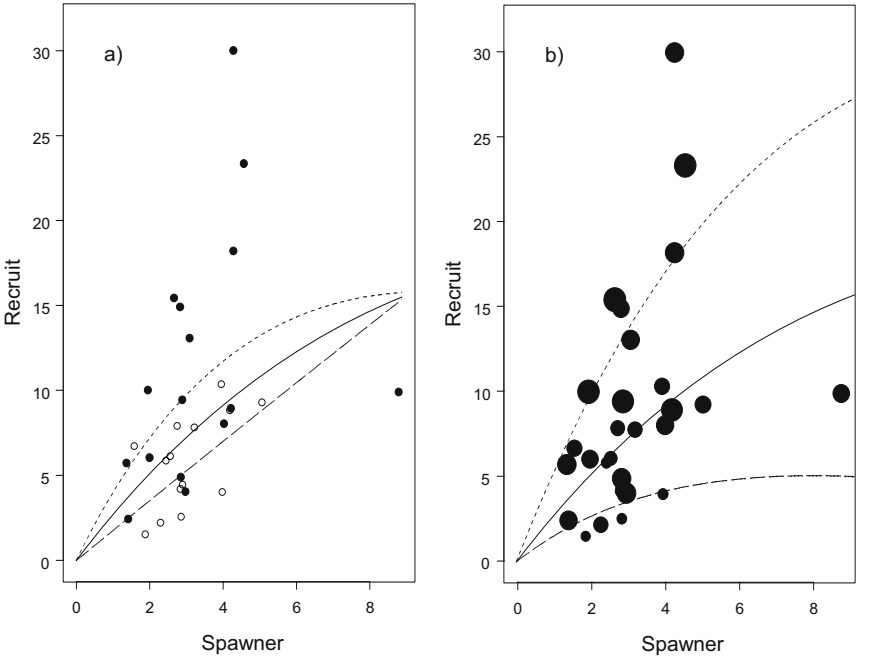


Figure 19.5. a) is for the Crisp-SR model. The bullets (•) are the SR data corresponding to the "Warm Years" and the circles (o) are the SR data corresponding to the "Cool Years". The lines from the top to the bottom are the fitted lines from Ricker model for the "Cool Years", all data combined and "Warm Years". b) is for the Fuzzy-SR model. The bullets (•) are the SR data with the radius in proportion to the magnitude of SST in that the higher the SST, the larger the radius. The lines in the top and the bottom are the Fuzzy-SR model fitted lines and the line in the middle is the fit from the simple Ricker model to all data combined. In both plots, the SR data is in unit of 1000 fish.

19.4.2.2 Fuzzy-SR Model Analysis

The same procedure described in Section 19.4.1.2 is carried out for these data. It was found that the estimate for β is close to zero from the hybrid optimal learning algorithm, which leads to a simplified FMF defined as $w_t = FMF_{Warm}(SST) = SST$ for the standardized SST (Fig.19.2b). For this FMF, there are no FMF parameters associated with it. Then the learning algorithm discussed in Section (19.3.1) is in fact the linear LSE, which is a global optimization to estimate the fuzzy SR

parameters (a_1, a_2, b_1 and b_2). Figure 19.5b) illustrates the performance of the Fuzzy-SR model and the estimated parameters as well as the summary statistics are summarized in Table 19.2. The Fuzzy-SR model produced the best fit to the original recruitment time series based on the estimated root-mean-square-error (RMSE), AIC and the correlation coefficient (r). The Crisp-SR model performed better than the Ricker-SR model.

19.4.2.3 Bootstrap Re-sampling Analysis

The residual diagnostics do not show any violation for the assumption of independence and homogeneity. Then the *bootstrap residuals* procedure in Section (19.3.2) is carried out for N = 1000 times to generate the bootstrap samples for the Fuzzy-SR parameters (a_1, a_2, b_1 and b_2). These bootstrap samples can be readily used to obtain the uncertainty estimate, such as confidence intervals and standard errors (Table 19.2 and Table 19.3). In addition, these samples can be used to test the significance of environmental impact (SST) on this stock (the first row in Fig. 19.6). It can be concluded that the SST has highly significant impact on this stock and the productivity parameter increased from 0.52 in "Cool" regime to 1.72 in "Warm" regime.

Furthermore, the associated management policy parameters S_{MSY} and μ_{MSY} can be readily calculated from equation (19.4) based on the bootstrap samples (Table 19.3). The resultant sampling distributions are illustrated in Fig. 19.6 (last two rows). It is also apparent that the distributions for these parameters are not exactly normal.

19.5 Summary and Discussion

The Fuzzy-SR model developed in this paper for SR analysis was based on extensions of the traditional Ricker model (Ricker-SR) and the Ricker model with crisp classification for the selected environmental variable (Crisp-SR). This approach can be naturally extended to any other form of SR models, such as the Beverton-Holt; Cushing; Deriso-Schnute and Shepherd listed in Quinn and Deriso (1999). Although the Fuzzy-SR model in this paper was illustrated by only one environmental variable (i.e. SST), it can be easily adapted to classify more environmental factors (such as salinity) and any fishery intervention factors. Unlike traditional SR models, Fuzzy-SR adapts the fuzzy logic decision algorithm, which helps to classify underlying empirical relationships. This enables more reasonable parameter estimates and consequently better advice for fisheries management. To address the lack of suitable uncertainty estimation in the fuzzy logic machine-learning method, a bootstrap re-sampling approach was proposed to make statistical inference for the SR parameters, to develop distribution plots for

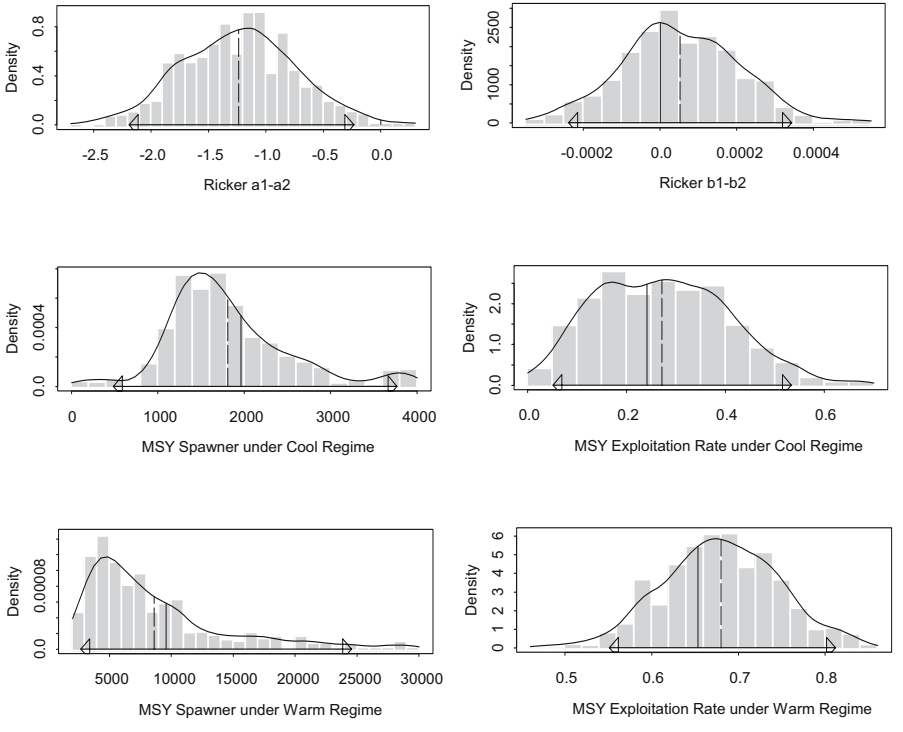


Figure 19.6. Bootstrapping sampling distributions for a_1 - a_2, b_1 - b_2, S_{MSY} and μ_{MSY} for "Cool" and "Warm" regimes. In each plot, the line at the top of the histogram bars is the kernel density estimate of the probability density function and the vertical dashed line is the sample mean. The horizontal lines with open arrows in the end denote the 95% sample confidence interval. In the first row, the vertical sold lines from zero is to use to test whether the difference between the two parameters is statistically significant. In the last two rows, the vertical sold lines are the parameter estimates from the Fuzzy-SR model.

the SR parameters productivity and capacity (i.e. a and b), and further to make inferences for fishery policy parameters. It was found that the resampling distributions for the stock parameters did not appear to be exactly normally distributed and therefore the approaches from Ricker-SR and Crisp-SR normally used to estimate SR parameters may not be appropriate. This serves as a warning for the use of simple regression statistics in SR analysis.

This Fuzzy-SR model is evaluated for two independent sets of fish stocks. The model is capable of classifying the effects of environmental interventions on the SR process based on the fuzzy logic algorithms. Fuzzy logic operations are used to categorize the input/output information on environmental intervention (SST) into

fuzzy sets with an associated degree of membership function based on Takagi and Sugeno (1983). The inherent uncertainties in the environmental data were taken into account by the fuzzification process. The Fuzzy-SR model is capable of empirically approximating the underlying SR relationship, and can also provide a crisp and simple functional relationship among the inputs and output according to the fuzzy rules (two in this application). An important feature of the Fuzzy-SR model is that the functional SR relationships described by the fuzzy rules can be chosen to more realistically describe the biological processes that affect recruitment.

Accordingly, the Fuzzy-SR model with the bootstrap resampling algorithm can be a useful tool for stock recruitment analysis to fish population.

Acknowledgements

I sincerely thank Jim Irvine, Jake Schweighert and Michael Folkes for their constructive suggestions and comments for this paper.

References

Bandemer H, Gottwald S (1995) Fuzzy Sets, Fuzzy Logic, Fuzzy Methods with Applications. John Wiley & Sons

Box GEP, Jenkins GM, Reinsel GC (1994) "Time Series Analysis: Forecasting and Control", 3rd Edition, Holden-Day

Chen DG, Ware DW (1999) A neural network model for forecasting fish stock recruitment. Can. J. Fish. Aquat. Sci. 56:2385-2396

Chen DG, Hargreaves B, Ware DM, Liu Y (2000) A fuzzy logic model with genetic algorithms for analyzing fish stock-recruitment relationships. Can. J. Fish. Aquat. Sci. 57:1878-1887

Chen DG, Irvine JR (2001) A new semiparametric model to examine stock-recruitment relationships incorporating environmental data. Can. J. Fish. Aquat. Sci. 58:1178-1186

Davison AC, Hinkley DV (1997) Bootstrap Methods and Their Application. Cambridge University Press

Efron B, Tibshirani RJ (1993) An Introduction to the Bootstrap. San Francisco: Chapman & Hall

Goodwin GC, Sin KS (1984) *Adaptive Filtering Prediction and Control*. Prentice-Hall, Englewood Cliffs, N.J

Hilborn R (1985) Simplified calculation of optimum spawning stock size from Ricker's stock recruitment curve. Can. J. Fish. Aquat. Sci. 42:1833-1834

Hilborn R, Walters CJ (1992) Quantitative Fisheries Stock Assessment: Choice, Dynamics and Uncertainty. Chapmam & Hall

Hyatt KD, Luedke W, Rankin DP, Gordon L (1994) Review of 1988-1994 forecast performance, stock status, and 1995 forecasts of Barkley Sound sockeye. Pacific Stock Assessment Review Committee. Working paper S94-21, p43

Koslow JA, Thompson KR, Silvert W (1986) Recruitment to Northwest Atlantic Cod (*Gadus morhua*) and Haddock (*Melanogrammus aeglefinus*) stocks: influence of stock size and climate. Can. J. Fish. Aquat. Sci., Vol 44, 26-39

Lee CC (1990) Fuzzy logic in control systems: fuzzy logic controller. *IEEE* Trans. On Systems, Man, and Cybernetics, 20(2): 419-435

Lek S, Belaud A, Dimopoulos I, Lauga J, Moreau J (1995) Improved estimation, using neural networks, of the food consumption of fish populations. Mar. Freshwater Res., 46, 1229-1236

Mackinson S, Vasconcellos M, Newlands N (1999) A new approach to the analysis of stock-recruitment relationships: "model-free estimation" using fuzzy logic. Can. J. Fish. Aquat. Sci. 56:686-699

Quinn II TJ, Deriso RB (1999) Quantitative Fish Dynamics. New York • Oxford. Oxford University Press

Ryall P, Murray C, Palermo V, Bailey D, Chen DG (1999) Status of Clockwork Chum salmon stock and review of the clockwork management strategy. Canadian Stock Assessment Secretariat Research Document 99/169 (ISSN 1480-4883), pp134. (http://www.dfo-mpo.gc.ca/csas/csas/resdoc/1999/index99.html)

Press WH, Flannery BP, Teukolsky SA, Vetterling WT (1988) Numerical Recipes in C: The Art of Scientific Computing. Cambridge University Press, Cambridge

Ricker WE (1975) Computation and Interpretation of Biological Statistics of Fish Population. Fisheries Research Board of Canada, Bulletin No. 191

Sakamoto Y, Ishiguro M, Kitagawa G (1986) Akaike Information Criterion Statistics, D. Reidel Publishing Company

Salia SB (1996) Guide to some computerised artificial intelligence methods. *In* Computers in Fisheries Research *Edited by* B.A. Megrey and E.Moksness, pp8-40. London:Chapman and Hall

Schweigert JF, Noakes DJ (1990) Forecasting Pacific herring (*Clupea harengus pallasi*) recruitment from spawner abundance and environmental information. Proc. Int. Herring Symposium, Oct. 1990, Anchorage, Alaska, 373-387

Shao J, Tu D (1995) The Jackknife and Bootstrap. New York: Springer-Verlag. Strobach P (1990) *Linear Prediction Theory: A Mathematical Basis for Adaptive Systems.* Springer-Verlag

Takagi T, Sugeno M (1983) Derivation of fuzzy control rules from human operator's control actions. *Proc. Of the IFAC Symp. On Fuzzy Information, Knowledge Representation and Decision Analysis*, Pages 55-60, July 1983

Tang B, Hsieh WW, Monahan AH, Tangang F (2000) Skill comparisons between neural networks and canonical correlation analysis in prediction the equatorial Pacific sea surface temperature. Journal of Climate. 287-293

Tester AL (1948) The efficacy of catch limitations in regulating the British Columbia herring fishery. Trans. Roy. Soc. Canada, 62:135-163

Tibshirani R (1994) A comparison of some error estimates for neural network models. Technical working paper No. 94-10, Department of Statistics, University of Toronto

Ware DM (1991) Climate, predator and prey: behavior of a linked oscillating system. Long-term variability of pelagic fish populations and their environment

T.Kawasaki et al.(Eds.). *1991. Pergamon Press. Tokyo. 402 pp*., 279-291

Ware DM (1996) Herring carrying capacity and sustainable harvest rates in different climate regimes. Pac. Stock Ass. Rev. Com. Working Paper H96-3, 19 p

Ware DM, McFarlane GA (1995) Climate-induced changes in Pacific hake (*Merluccius productus*) abundance and pelagic community interactions in the Vancouver Island upwelling system. *In* Climate change and northern fish populations, *Edited by* R. J. Beamish. Canadian Special Publication of Fisheries and Aquatic Sciences 121, 509-521

20. Computational Assemblage of Ordinary Differential Equations for Chlorophyll-*a* Using a Lake Process Equation Library and Measured Data of Lake Kasumigaura

N. Atanasova · F. Recknagel · L. Todorovski · S. Džeroski · B. Kompare

20.1 Introduction

Lake ecosystems are highly complex dynamic systems. Modelling of such ecosystems is ongoing challenges scientists, who continue to gain better understanding of ecological processes in order to more realistic simulate ecosystem behaviours. Two basic modelling approaches can be distinguished: the deductive, knowledge driven approach resulting in deterministic models, and the inductive, data driven approach exploring candidate models and match them with measured data resulting in empirical models.

Deterministic models are typically represented by ordinary differential equations (ODE) which are being applied to lake ecosystems since the 1970s (e.g. Straskraba and Gnauck 1984; Recknagel 1989; De Angelis 1992; Chapra 1997; Jorgensen and Bendoricchio 2001). If applied to real lake data ODE can be well adjusted and interpreted in the context of the domain due to their explicit causality. However, complex ecological processes are often not yet fully understood and therefore ODE are sometimes adapted to our incomplete knowledge resulting in simplified models.

By contrast inductive models induced from the data by bio-inspired computation such as artificial neutral networks and evolutionary algorithms may rely heavily on the comprehensiveness of data. They have been demonstrated to be powerful predictive tools (e.g. Recknagel et al. 2002; Lee et al. 2005) but may still be limited in their representation and explanation.

In this paper we apply an approach that combines both domain knowledge and data. The domain knowledge is gathered in a knowledge library, which is used to guide the process of induction from real data. The result is a set of elementary process descriptions for ODE that match basic principles of the domain of interest (Todorovski and Dzeroski, 2001; Langley et Al., 2002; Todorovski, 2003). In the early days of the development of these tools (Todorovski & Džeroski, 1997), the knowledge had to be provided as an explicit definition of the space of candidate models. Now, these tools allow the user to provide higher-level domain knowledge about building mathematical models of complex real-world systems.

In this paper we apply the combined modelling approach to Lake Kasumigaura (Japan) by utilising a library for process equations of lake domain knowledge and measured data. Previous research on modelling of lake Kasumigaura was based on artificial neural networks (ANN), genetic algorithms (GA) and evolutionary algorithms (EA). ANN was trained to predict the dominant algal genera (Recknagel et al. 1997; Recknagel et al. 1998; Wei et. al., 2001) and zooplankton abundance (Recknagel et al. 1998) in Lake Kasumigaura. GA was applied to induce predictive ODE for Chl-a (Whigham and Recknagel 2001) and EA to induce predictive rules for Chl-a in the lake (Bobbin and Recknagel 2001; Recknagel et al. 2002). In the context of this research we attempt to discover predictive ODE for the Chl-a by assembling and adapting process equations from a lake domain library.

20.2 Methods and Material

20.2.1 LAGRAMGE: Computational Assemblage of ODE

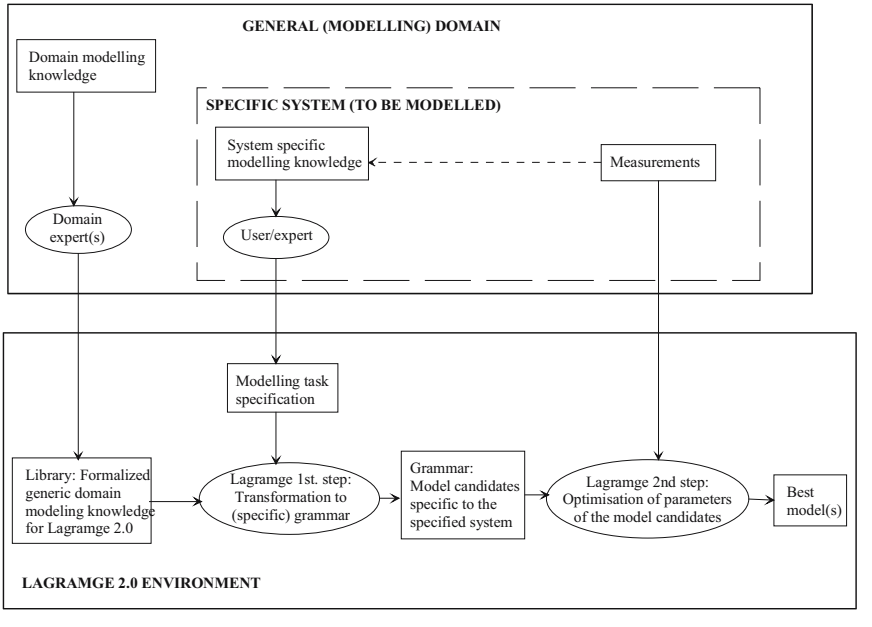

GENERAL (MODELLING) DOMAIN
Domain modelling knowledge
SPECIFIC SYSTEM (TO BE MODELLED)
System specific modelling knowledge
Measurements
Domain expert(s)
User/expert
Modelling task specification
Library: Formalized generic domain modeling knowledge for Lagramge 2.0
Lagramge 1st. step: Transformation to (specific) grammar
Grammar: Model candidates specific to the specified system
Lagramge 2nd step: Optimisation of parameters of the model candidates
Best model(s)
LAGRAMGE 2.0 ENVIRONMENT

Fig. 20.1. An automated modeling framework based on the integration of domain-specific modeling knowledge in the process of equation discovery

The principal concept of computational assemblage of ODE by LAGRAMGE is shown in Fig. 20.1. After the modelling task has been defined and the lake data been specified domain knowledge is transformed from the library into a grammar. This grammar specifies the space of candidate models as illustrated in the left part side of Fig. 20.1. Once the grammar has been determined, LAGRAMGE is heuristically searching through the space of candidate models and testing each of them with measured data after fitting constant parameter values. These models are evaluated by means of two error measurements. One is mean square error (MSE) – it measures the discrepancy between measured data and data obtained by simulating the model. The other is minimum description length (MDL) function that takes into account model complexity. The function contains an additional term that introduces a penalty for the complexity of the equation. Further details about the algorithm of LAGRAMGE can be found in (Todorovski, 2003).

20.2.2 Domain Knowledge Library for Lake Ecosystems

In order to be used in the model induction procedure, the knowledge needs to be coded in the knowledge library. Todorovski (2003) developed the formalism for encoding the domain knowledge about lake ecosystems. Using this formalism Atanasova et al. (2004) developed a comprehensive knowledge library for lakes ecosystems. The library supports the construction of 0-dimensional N-box models, i.e., supports modelling of stratified lakes. The equations coded in the library are recruited from literature models developed for lakes, and can be assembled to different levels of ecosystem structures such as the simple Vollenweider model (Vollenweider, 1968) or the fairly complex model SALMO (Benndorf and Recknagel 1982; Recknagel and Benndorf 1982). For more details see Atanasova (2004).

In general, the knowledge coded in the library can be conceptually presented as shown in Fig. 20.2, where only a part of the library is depicted. The boxes represent the types of state variables, whereas the arrows stand for ecological processes that influence the state variables. According to this diagram the library allows for modelling of dissolved inorganic nutrients (e.g. inorganic nitrogen, phosphorus and silica), primary producers (e.g. diatoms and green algae), secondary producers (e.g. zooplankton), dissolved organic matter and detritus. Processes, which are in the library, but not depicted on Fig. 20.2 are describing dissolved oxygen pathways such as aeration, oxygen production or consumption processes.

The knowledge in the library is formalized in terms of the: (1) taxonomy of variable types, (2) taxonomy of basic processes that govern the behavior of the state variables, (3) alternative models of the basic processes, and (4) knowledge how to combine models of individual processes to a system of ODE for an ecosystem.

Basic processes (arrows in Fig. 20.2) are declared as *process classes*. A process class represents different formulations of a certain basic process. For example, the

process that describes a primary producer growth (arrow no. 1 in Fig. 20.2) includes the exponential, logistic and limited growth models. Furthermore, the limited growth model includes different formulations growth functions limited by nutrients, light or temperature.

According to the ODE for the state variables the classes of processes are combined by so-called *combining schemes*. Combining scheme of specific variable represent the all processes that may affect that variable. In other words, each combining scheme represents a differential equation for the specific state variable. Thus, the library contains six combining schemes for six dependant (state) variable types.

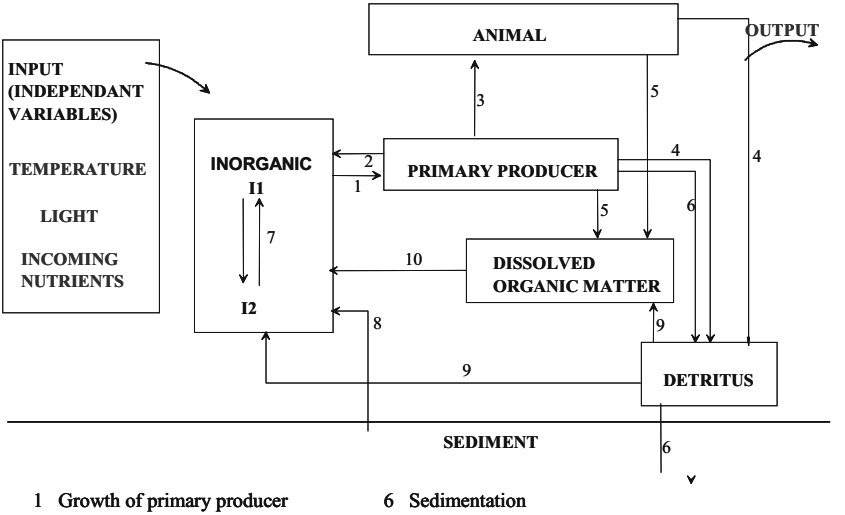


Fig. 20.2. Generalized scheme of compartments and interactions

20.2.2 Task Specification

The domain knowledge library comprises general knowledge about modelling of lakes. In the task specification the user of LAGRAMGE provides the specific knowledge and data of the lake to be modelled. It includes the selection of variables and processes. The model variables are specified by the variable type and the variable name as follows:

variable variable_type 'variable_name'

The word **system** in front of the word **variable** variable specifies a state variable. The process variables are defined by the word ***process***, followed by the process name and the process arguments:

process 'process name'(argument1, argument2, ...) **process_notation**

Arguments represent the variables in the observed system that influence (or are influenced by) the specific process. They are used in the process formulations in the library. If some of the arguments in the process are considered as sets within the process then we put the names of those arguments into brackets {}. A set can contain none (empty), one or many variables (arguments) of the same type.

Tab. 20.1. Declared variable types in the knowledge library

Variable type	Description	dependant (state) / independent (forcing)
type Concentration **is real**	concentration of a substance	generic
type Light **is real**	light intensity	independent
type Temperature **is real**	temperature	independent
type Precipitation **is real**	precipitations	independent
type Flow **is real**	flow rate	independent
type Area **is real**	contributing area of the incoming nutrients	independent
type Inorganic **is Concentration**	dissolved inorganic nutrients	dependant
type Population **is Concentration**	concentration of a population	generic
type Detritus **is Population**	particulate dead organic matter	dependant
type Oxygen **is Concentration**	dissolved oxygen	dependent
type Dom **is Concentration**	dissolved organic matter	dependant
type Primary_producer **is Population**	primary producers	dependant
type Animal **is Population**	secondary producers	dependant

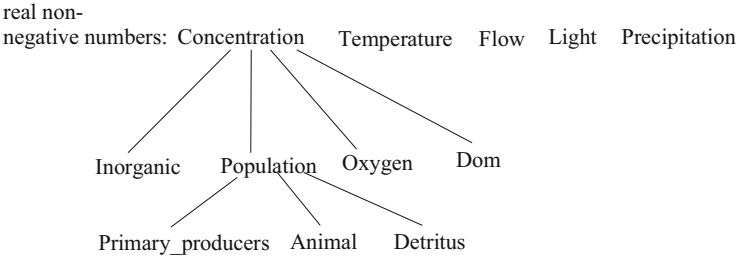


Fig. 20.3. Graphical presentation of variable types and sub-types in the knowledge library for lake modelling

Thus, in order to correctly introduce the expert knowledge to LAGRAMGE we need to know the: (1) types of variables declared in the library and (2) types of ecological processes declared in the knowledge library. The types of the variables in the knowledge library for lakes are given in Tab. 20.1. The type *Concentration* is a generic variable type that is determined by sub-types of variables. It has four sub-types, i.e. *Inorganic* representing the dissolved inorganic nutrients, *Population* representing particulate organic matter, *Dom* denoting a dissolved organic matter and *Oxygen* representing dissolved oxygen concentration. Population has again three sub-types – *Primary_producers*, *Animal* and *Detritus*. The types of variables are schematically shown in Fig. 20.3.

Tab. 20.2. Description of process' definition in the knowledge library

	Process description	Process name	Arguments: types of variables involved in the process' formulations	Argument declared as Set: y/n
1	Outflow of a substance from the system	Outflow	1. Concentration 2. Flow	n n
2	Inflow of a substance to the system	Inflow	1. Concentration 2. Concentration 3. Flow	n n n
3	Settling of a substance	Sedimentation	1. Concentration 2. Temperature	n y
4	Diffusion	Diffusion	1. Concentration 2. Concentration	n n
5	Growth of a primary producer	PP_growth	1. Primary_producer 2. Inorganic 3. Temperature 4. Light	n y y y
6	Predator prey interactions	Feeds_on	1. Animal 2. Population 3. Temperature	n y y
7	Respiration of a primary producer	Respiration_PP	1. Primary_producer 2. Inorganics 3. Temperature 4. Light	n y y y
8	Respiration of an animal (sec. prod)	Respiration_A	1. Animal 2. Temperature	n y
9	Natural mortality of a primary producer	Mortality_PP	1. Primary_producer 2. Inorganic 3. Temperature 4. Light	n y y y
10	Natural mortality of an animal (sec. prod)	Mortality_A	1. Animal 2. Temperature	n y
11	Excretion from secondary producers	Excretion_A	1. Animal 2. Temperature	n y

If we want to model interactions between several species (for example primary producer grazing on more then one nutrient) we need to declare sets of variables. Declaration of set Primary_producers of the type Primary producer is given below.

Please note the plural form of the set name, which is derived from the singular name of the variable type name:

type Primary_producers **is set(**Primary_producer**).**

The Tab. 20.2 includes the description of the majority of the processes declared in the knowledge library. In the first column the description of the ecological processes is given. The second column contains the processes' names as they are declared in the library. The third and the fourth column give information about the arguments, i.e. the variables involved in the processes' formulations. In the third column the types of the involved variables (arguments) are listed, whereas the fourth contains information whether the variable is included in the process declaration as set or not.

For example, in line 5 the definition of the ecological process Growth of a primary producer is given. The process name is PP_growth and it has 4 arguments. The first is of type Primary_producer and it represents the variable which the process refers to. The rest of the arguments are variables of types Inorganic, Temperature and Light. They are all declared in the library as sets. The statement in the task specification ***process*** PP_growth(phyto1, {ps}, {temp}, {llight}) **growth,** describes the growth of a primary producer *phyto1*. The process is influenced by single inorganic nutrient *ps,* temperature *temp* and light *light* respectively. Living one of the brackets {} empty would indicate no influence by the variable which was left out. For instance definition of a growth process of phytoplankton (phyto1) that is influenced by two nutrients phosphorus (ps) and nitrogen (ns) and temperature (temp), but not light (light) limited would be:

process PP_growth(phyto1, {ps, ns}, {temp}, {}) **growth.**

Note that this specific "Lake" knowledge library includes several formulations for each of the process classes in the task specification (Atanasova et al., 2004). For example the process class PP_growth contains five different models for primary producer growth, i.e. exponential, logistic, growth limited by temperature, light and nutrients, growth limited model that accounts for variable optimal temperature and growth limited model that couples the effects of light and temperature. Furthermore, light, temperature and nutrients limitations are defined as function classes that include several different formulations for each. Thus, we have more than fifty possible formulations for the PP_growth process, which are all correct from the standpoint of the used library and defined task. Similarly, we have several possible formulations for the rest of the process classes in this system.

In order to find a model of a specific system with Lagramge we need (1) measurements of the state (dependent) and forcing (independent) variables that will be used in the optimisation procedure and (2) expert knowledge about the variables and processes, which will be used for determining the model structure.

20.2.3
Data of Lake Kasumigaura

Lake Kasumigaura is a shallow lake in Japan with maximal depth of 7 m and average depth of 4 m. It has a volume of 662 million m^3 and a surface area of 220 km^2. The hypereutrophic state of the lake causes blue-green algal blooms in summer and autumn with frequently high abundances of *Microcystis* and *Oscillatoria.* The Tab. 20.3 summarises the measured data of Lake Kasumigaura from 1986 to 1992 that were used as in a daily interpolated format in the context of this study.

Tab. 20.3. Structure of the database of Lake Kasumigaura for 1986 to 1992

Limnological Variables	**Mean / Min / Max**
PO_4 μg/l	14.16 / 1 / 235
NO_3 mg/l	0.52 / 0.001 / 2.39
Si mg/l	3.29 / 0.015 / 12.49
Chl*a* μg/l	74.5 / 0.69 / 279.5
Water Temperature °C (WT)	16.37 / 2.1 / 32
Solar Radiation Jcm^{-2} day^{-1}	1281 / 65 / 3364
Phytoplankton cells/ml *Microcystis* and *Oscillatoria* *Scenedesmus* *Synedra* Zooplankton individuals/l *Cladocera*	 28735 / 0 / 616666 and 17765 / 0 / 250775 833 / 0 / 11648 4990 / 0 / 75130 157 / 0 / 1002

20.2.4
Experimental Framework

In order to test the performance of the LAGRAMGE algorithm for the simulation of chlorophyll-a (chl-*a*) by means of ODE assembled and adapted to data from Lake Kasumigaura following experiments were designed and conducted:

- Experiment 1: Discover chl-*a* models for each year separately. This experiment focused on the question whether it is possible to find a generic model structure for all years from 1986 to 1992 and just optimise the parameter values for each year or to require specific model structures for each year. We tested each year-specific model on the remaining years in order to find out whether there is a generic model for all measured years. Algal grazing by zooplankton was not included in this experiment as zooplankton data were only available for the years 1986 to 1989.
- Experiment 2: Discover one chl-*a* model for all years from 1986 to 1992. This experiment focused on the question whether it is possible to derive a generic model from all data that would be valid for each single year. The model was trained by data from 1986 to 1991, and tested for the year 1992. Algal grazing by zooplankton was not included in this experiment as zooplankton data were only

available for the years 1986 to 1989.
- Experiment 3: Discover one chl-*a* model including algal grazing by zooplankton by using the years 1986 to 1988 for learning and 1989 for testing.

The task specification for experiment (3) is given in Tab. 20.4. Following types of variables are declared: inorganic nutrients, i.e nitrogen_nitrate (*no3*), dissolved inorganic phosphorus (*ps*) and silica (*silica*), primary producer (*chla*), animal (*clad*), temperature (*temp*) and light (*light*). The word **system** in front of the primary producer declaration denotes that only *chla* model will be discovered (*chla* is the only state variable), while the rest of the variables will be considered as independent variables. The processes are declared in lines from 8 to 11. Phytoplankton growth is described in line 8 (recall the process description from the previous section). The process Feeds_on (line 9) stands for (1) predatory loss of phytoplankton (chla) and (2) growth of zooplankton (clad). Optional arguments of this process are the food (phyto) and temperature (temp), which means that the growth of clad can be or not influenced by the food (none or many species) and temperature. Similarly the rest of the processes in the system (Respiration_PP, and Sedimentation) are defined (see lines 10 and 11).

Tab. 20.4. Modelling task specification for lake Kasumigaura

1:	variable Inorganic ps
2:	variable Inorganic no3
3:	variable Inorganic silica
4:	system variable Primary_producer chla
5:	variable Animal clad
6:	variable Temperature temp
7:	variable Light light
8:	process PP_growth(chla, {ps, no3, silica}, {temp}, {light}) gr1
9:	process Feeds_on(clad, {chla}, {temp}) feeds1
10:	process Respiration_PP(chla, {temp},{},{}) resp1
11:	process Sedimentation(chla, {temp}) sed1

According to the experimental setup the grazing process (Feeds_on) was either included or excluded from the induction procedure. The task specification from Tab. 20.4 was modified for this case by replacing the process Feeds_on by natural mortality (Mortality_PP): process Mortality_PP(chla, {temp}, {}, {}) mort1.

According to the combining schemes (mass balances) declared in the library, this task specification gives either the model structure as (20.1), or in the case of replacing the Feeds_on process (predatory loss) by natural mortality as (20.2):

$$\frac{dchla}{dt} = \text{PP\_growth - Respiration - Sedimentation - Feeds\_on} \qquad (20.1)$$

$$\frac{dchla}{dt} = \text{PP\_growth - Respiration\_PP - Mortality\_PP - Sedimentation} \qquad (20.2)$$

Note that the formulation of the process loss of phytoplankton by grazing needed some adjustments since the zooplankton abundance unit [individuals/l] was not compatible with the biomass unit [mass/volume]. We overcame this problem by allowing only one possible formulation of the Feeds_on process in the knowledge library, i.e:

$$\mathrm{Feeds\_on\ (Grazing)} = C_{f\max} \cdot f1(temp) \cdot f2(F_T) \cdot clad \cdot chl\_a$$

where C_f is zooplankton filtration rate [ml/(individuals*time)], *clad* is the abundance of cladocera in [individuals/ml], *chl_a* is chlorophyll-*a* concentration in [mg/l chl-*a*], f1(temp) is temperature influence function (unitless) and $f(F_T)$ is food limitation function for zooplankton growth (unitless). In this case F_T represents the total phytoplankton concentration. Considering this, the loss of phytoplankton is calculated in [mg/l chl-*a*].

20.3 Results and Discussion

20.3.1 Experiment 1

This experiment aimed to identify separate ODE models for the calculation of chl-*a* for each years. Thus the LAGRAMGE algorithm discovered 7 models with corresponding MSE and MDL function. Due to the preference to the simpler models those with the minimal MDL values for each year were chosen as best models, i.e. equation (20.3) was the best model for 1986, equation (20.4) for 1987, equation (20.5) for 1988, equation (20.6) for 1989, equation (20.7) for 1990, equation (20.8) for 1991 and equation (20.9) for 1992:

$$\frac{dchla}{dt} = chla \cdot 0.152 \cdot \frac{ps}{ps+0} \cdot \frac{no3^2}{no3^2+4.7E-7} \cdot \frac{silica^2}{silica^2+0.011} \cdot \frac{temp-0}{15-5} \cdot \frac{light}{light+196.7} - chla \cdot 0.1 \cdot \frac{temp-5}{17.4-2.5} - chla \cdot chla \cdot 0.001 - chla \cdot \frac{0.04}{5} \tag{20.3}$$

$$\frac{dchla}{dt} = chla \cdot 0.08 \cdot \frac{ps^2}{ps^2+3.2E-6} \cdot \frac{no3}{no3+0.00012} \cdot \frac{silica^2}{silica^2+0.023} \cdot \frac{temp}{16.2} \cdot \frac{light}{light+41.8} - chla \cdot 0.005 - chla \cdot 0.01 \frac{temp-0}{15-5} - chla \cdot \frac{0.096}{5} \cdot 1.11^{(tmp-15)} \tag{20.4}$$

$$\frac{dchla}{dt} = chla \cdot 0.09 \cdot \frac{ps}{ps+0} \cdot \frac{no3}{no3+0} \cdot \frac{silica}{silica+0.022} \cdot \frac{temp}{10.8} \cdot \frac{light}{light+200} - chla \cdot 0.022 \cdot 1.11^{(temp-18.8)} - \\ - chla \cdot 0.01 \cdot \frac{temp}{7.2} - chla \cdot \frac{0.05}{5} \tag{20.5}$$

$$\frac{dchla}{dt} = chla \cdot 0.09 \cdot \frac{ps}{ps+0} \cdot \frac{no3}{no3+0} \cdot \frac{silica}{silica+0} \cdot \frac{temp}{6.4} \cdot \frac{light}{light+200} - chla \cdot 0.02 \cdot 1.13^{(temp-15)} - \\ chla \cdot chla \cdot 0.77 - chla \cdot \frac{0.14}{5} \tag{20.6}$$

$$\frac{dchla}{dt} = chla \cdot 0.134 \cdot \frac{ps}{ps+3.2E-5} \cdot \frac{no3}{no3+0} \cdot \frac{silica}{silica+0} \cdot \frac{temp}{19.8} \cdot \frac{light}{light+0} - chla \cdot 0.004 \cdot 1.12^{(temp-20)} - \\ chla \cdot chla \cdot 0.54 - chla \cdot \frac{0.28}{5} \cdot \frac{temp-5}{15-5} \tag{20.7}$$

$$\frac{dchla}{dt} = chla \cdot 0.224 \cdot \frac{ps}{ps+0} \cdot \frac{no3}{no3+0} \cdot \frac{silica}{silica+0} \cdot \frac{temp}{20} \cdot \frac{light}{light+10.3} - chla \cdot 0.0009 - \\ chla \cdot chla \cdot 0.332 - chla \cdot \frac{0.5}{5} \cdot \frac{temp-2}{15-5} \tag{20.8}$$

$$\frac{dchla}{dt} = chla \cdot 0.139 \cdot \frac{ps}{ps+0} \cdot \frac{no3}{no3+0} \cdot \frac{silica}{silica+0} \cdot 1.11^{(temp-19)} \cdot light \cdot e^{-\frac{\left(\frac{light}{179.5}+1\right)}{184.55}} - chla \cdot 0.056 \cdot 1.12^{(temp-15)} - \\ chla \cdot chla \cdot 0.023 \cdot \frac{temp}{1.3} - chla \cdot \frac{0.0001}{5} \tag{20.9}$$

The alternative model structures include processes as shown in equation (20.2). In all cases the growth term is dependent on nutrient concentrations, water temperature and underwater light. Nutrient limitation functions for ps, no3 and silica are formulated with the two variations of Monod term, i.e. $f(x) = \frac{x}{x + \text{constant}}$ or $f(x) = \frac{x^2}{x^2 + \text{constant}}$. Note that the smaller the constant (half saturation coefficient) in the Monod term the smaller is the influence by x. For example, a term with saturation coefficient zero, i.e., $\frac{x}{x+0}$ is equal to 1, which means no limitation (influence) by x. From this we can reveal the nutrients' influence on the total phytoplankton growth and how the limiting nutrient(s) is changing with time. According to the models this influence is pretty unpredictable, which is probably a result of the variety of algae species, in the total phytoplankton. Phosphorus was found to be the limiting nutrient only in 1990, and in 1987 together with nitrate and silica. Also the saturation constant in the phosphorus limitation function is very small. Nitrogen was the limiting

nutrient in 1986 (together with silica) and 1987 (together with phosphorus and silica), while silica was limiting in 1986, 1987 and 1988. Nutrients did not limit the phytoplankton growth in 1989, 1991 and 1992. To find the limiting nutrient it is crucial (1) to know the load of the lake with the nutrients (external and internal) and (2) to estimate which algae will bloom most severely. The latest is partly revealed by the models. In 1986 nitrogen limitation can be related with the severe microcistis blooms. There were small amounts of diatoms, obviously limited by silica. It is surprisingly for the 1987 model that all nutrients were found as limiting, although there are no diatoms identified in this year. Severe blooms of diatoms in 1988 were limited by silica as revealed by equation (5). According to the discovered models the lake receives quite a lot of nutrients, since the nutrients were not limiting the growth in 1989 1991 and 1992, and in 1990 the limitation by phosphorus is negligible.

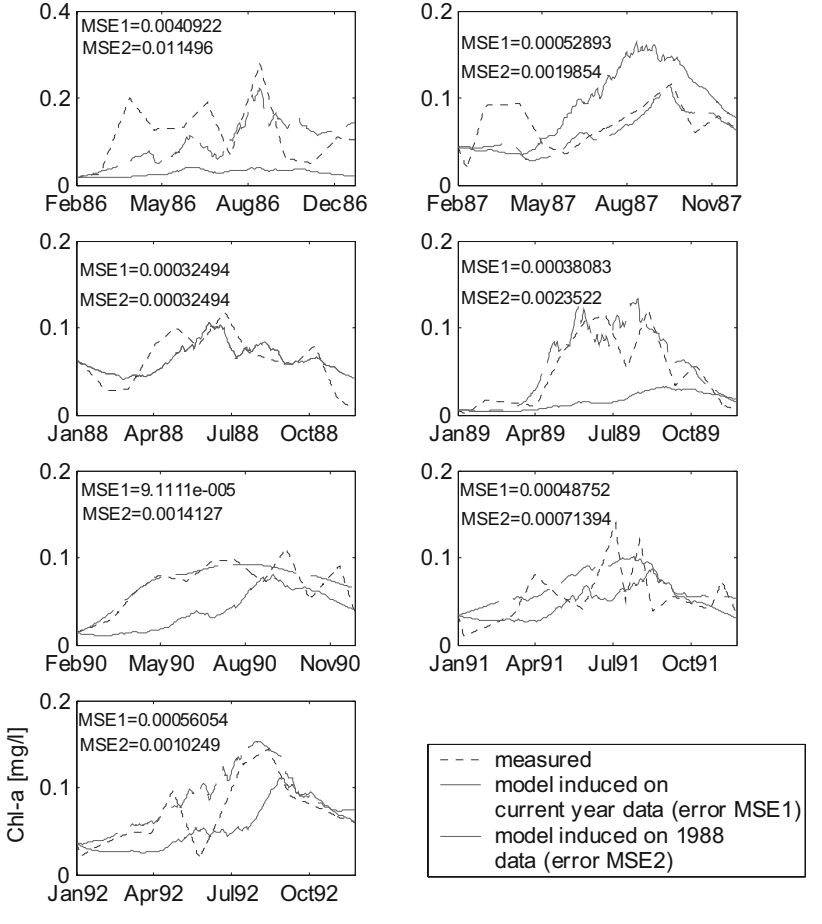


Fig. 20.4. Simulation results of the chl-*a* [mg/l] equations (20.3) to (20.9) assembled and trained by Lake Kasumigaura data of 1986 to 1992

Monod expression is used for light limitation function in all models except for 1992, where the photoinhibition formulation for light is used. Temperature influence is modelled with the linear temperature curve in all years except for 1992, when the influence is exponential. The rest of the processes, i.e. respiration, mortality and sedimentation are modelled with similar formulations in all models. The models differ greatly in the parameter values that may suggest that some of them should be replaced by variables.

The performance of the 6 models compared to the measured data is shown in Fig. 20.4. Most of the models are able to approximate well the timing and magnitude of chl-*a*.

In order to find a model that would simulate the phytoplankton during the entire period satisfactorily, each model was validated on the data set that was not used for training of that specific model. None of the discovered models could accurately simulate chl-*a* on unseen data, except for the equation (20.5) discovered for the data of 1988. The validation of this model is shown in Fig. 20.5. The model performs satisfactorily, except in 1986. This year seems to be quite unusual, since the chl-*a* peak is nearly twice as much as it is in the rest of the years.

20.3.2 Experiment 2

The first experiment has clearly demonstrated the highly dynamic nature of algal biomass represented in Lake Kasumigaura reflected by the calculated data of 6 annually specific ODE and the measured data for chl-*a*. Both vary distinctly in timing and magnitude year by year. To find a generic model structure that would accurately simulate chl-*a* dynamics for consecutive years is therefore very challenging. However the equation (20.5) discovered for the data of 1988 in the experiment 1 has indicated that LAGRAMGE can discover common patterns in complex data, and that the year 1988 provides average lake data which are suitable for training the chl-*a* model. Our second experiment aimed at the discovery of a generic chl-*a* model trained by data of all years 1986 to 1991. The ODE structure was specified according to equation (20.2). The ODE for chl-*a* with the lowest MDL identified by LAGRAMGE reads sa follows:

$$\frac{dchla}{dt} = chla \cdot 0.117 \cdot \frac{ps^2}{ps^2 + 2.6E-07} \cdot \frac{no3}{no3 + 9.8E-05} \cdot \frac{silica}{silica + 0} \cdot \frac{temp}{20} \cdot \frac{light}{light + 200} - chla \cdot 0.00658$$
$$- chla \cdot 0.003 \cdot \frac{temp}{3.3} - chla \cdot \frac{0.072}{5} \cdot 1.1^{(temp-18.1)} \tag{20.10}$$

This equation (20.10) reflects that nutrient concentrations are supposed to have little impact on the growth process as expressed by the modified Monod kinetics in the first term. Half saturation constants in phosphorus and nitrogen limiting functions were calibrated by LAGRAMGE with very small values, i.e. 2.6E-07 and 9.8E-05, whereas silica has no influence at all with half saturation constant 0.

The respiration process is formulated by simple first order kinetics. Mortality and sedimentation, i.e. the last two terms are formulated as temperature dependant processes. Equation (20.10) has a similar structure as equation (20.3) to (20.9) but different parameter values. However in contrast to the equation (20.5) discovered for the year 1988 it does not consider silica as limiting nutrient.

As the numerical solution of ODE requires initial values for each state variable we provided the measured initial values for the first day of each year in case that we simulated consecutive years as for experiment 2.

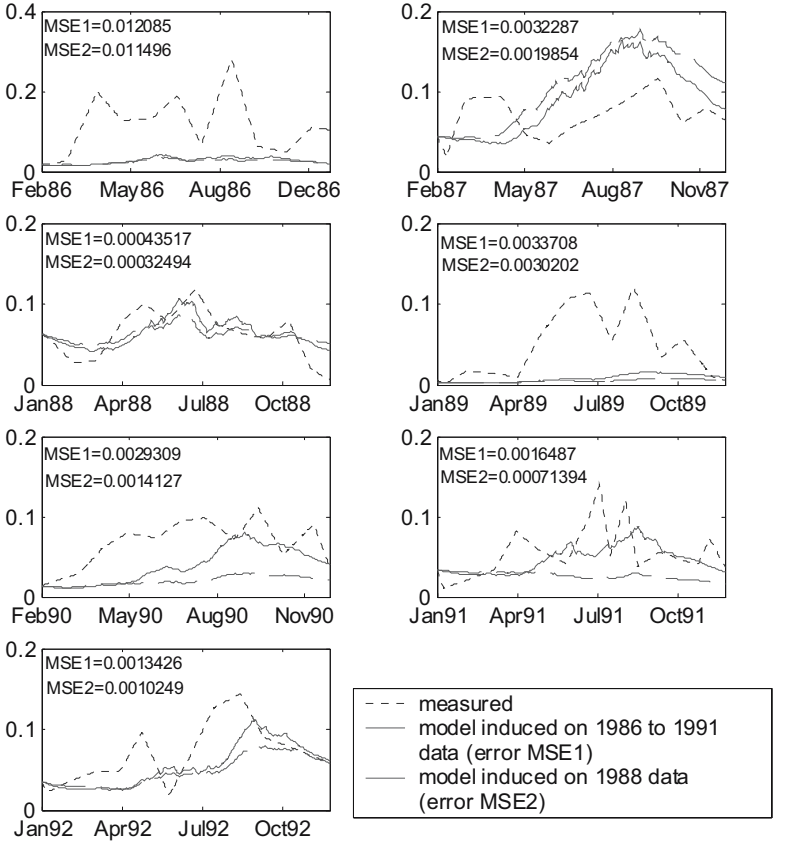


Fig. 20.5. Simulation results of the chl-*a* [mg/l] equation (20.10) assembled and trained by Lake Kasumigaura data of 1986 to 1992 and tested by the data of 1992 (dashed line) and equation (20.5) trained on 1988 data (solid line)

The Fig. 20.5 illustrates the simulation results of experiment 2 where the ODE structure and parameters for chl_*a* were assembled and adapted according to the Lake Kasumigaura data from 1986 to 1991, and tested for data of 1992. Though not very accurate the model still manages to predict most of the chl-a peaks and

crashes. The simulation is best for years 1987 and 1988 and least accurate for 1986. The model quite accurately performs on the unseen data, i.e. data from 1992 (see Fig. 20.5).

In comparison with the model discovered on 1988 this model did not perform so well, though it was trained on longer data set. Possible explanation of this is that there is more noise in the long data set so it is more difficult to learn the lake's behaviour (with the present optimisation method). On the other hand learning the behaviour from one year's data is much easier but the year should be representative enough so the model can be evaluated on longer period, which was the case in this study. In any case, long term data set is needed in order to draw some relevant conclusions.

20.3.3. Experiment 3

The experiment 3 was carried out with Lake Kasumigaura data from 1986 to 1988 for training and data of 1989 for testing by adding the grazing process to the ODE for chl_*a* according to the task specification in Tab. 20.4. As a result the equation (20.11) had been discovered with the lowest value of MDL:

$$\frac{dchla}{dt} = chla \cdot 0.107 \cdot \frac{ortp}{ortp + 4.7E - 10} \cdot \frac{no3}{no3 + 0.00016} \cdot \frac{silica^2}{silica^2 + 0.01} \cdot \frac{temp}{9.3} \cdot \frac{light}{light + 147.6} - chla \cdot 0.054 \cdot \frac{temp - 2.4}{15 - 5} - \\ - chla \cdot \frac{0.009}{5} \cdot \frac{temp}{4.6} - clad \cdot 0.12 \cdot \frac{temp}{9.53} \cdot \frac{chla}{chla + 0} \cdot chla \cdot 0.07 \tag{20.11}$$

In equation (20.11) the impact of nutrients on the growth process appears to be strengthened compared to equation (20.10). The grazing rate (Feeds_on) has been formulated by a proportional relationship with zooplankton (clad) and phytoplankton (chl_*a*) biomass as well as water temperature. The constant parameter value 0.07 indicates that only small amount of chl-a is consumed by zooplankton grazing. The training results of equation (20.11) achieve better MSE values for 1986 and 1989 (see Fig. 20.6) when compared with the previous equation (20.10). It captures the trend in 1987 but it overestimates the late summer peak as it similarly does in 1988. The equation (20.10) simulated the seasonal dynamics of chl_*a* in 1986 very poorly. It didn't simulate well the chl_a dynamics in 1987 and 1989 by underestimating the spring and early summer peaks of both years and overestimating the late summer peak in 1987.

From experiment 3 it can be concluded that LAGRAMGE could not assemble a reasonable chl_*a* equation from data of one year only that would accurately simulate chl_*a* of other years. However some better simulation results compared with those from the first experiment were achieved by chl_*a* equations assembled and trained separately by data of each year i.e. equation (20.12) for 1986, (20.13) for 1987, (20.14) for 1988 and (20.15) for 1989 (see Fig. 20.7). These models achieved fairly good simulation results for the years 1988 and 1989, but underestimated spring peaks in 1986 and 87. As expected the equations (20.12) to (20.15) show that rate functions for growth, respiration and sedimentation are

differently represented when grazing is added to the chl_a mass balance equations. As a result the growth rates consider differently limiting nutrients, i.e. in 1986 phosphorus is identified in addition to nitrogen and silica, in 1987 all three nutrients remain limiting, in 1988 nitrogen is added to silica, and in 1989 all nutrients are limiting.

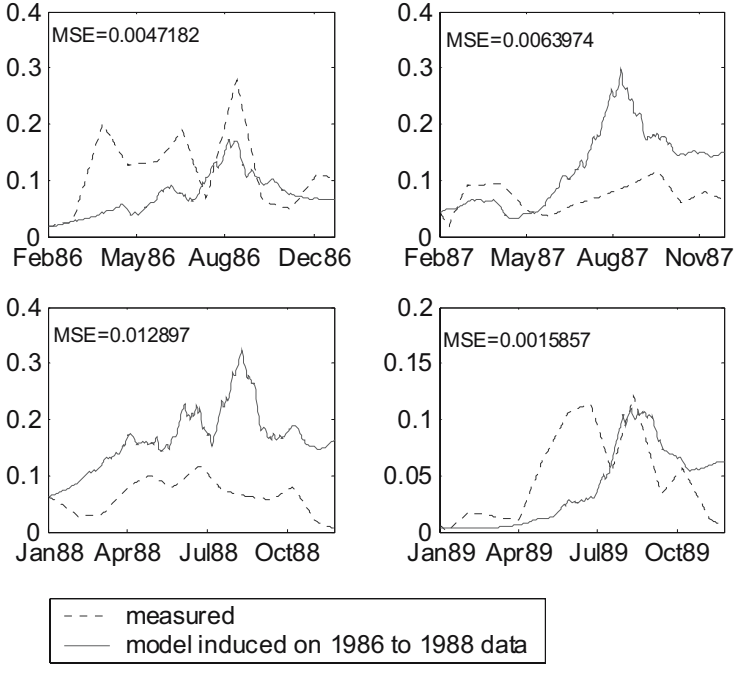


Fig. 20.6. Simulation results of the chl-*a* [mg/l] equation (20.11) annually assembled and trained by Lake Kasumigaura data of 1986 to 1988 and tested by the data of 1989

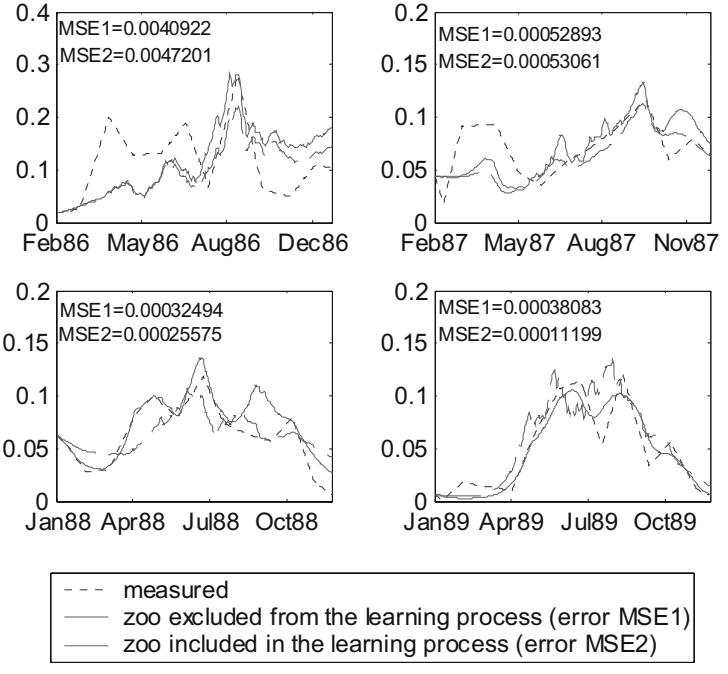


Fig. 20.7. Simulation results of the chl-*a* [mg/l] equations (20.12) to (20.15) annually assembled and trained by Lake Kasumigaura data of 1986 to 1989

These findings highlight the need to represent different functional algal groups such as diatoms, green and blue-green algae by separate ODE in order to realistically consider their specific nutrient requirements and selective preferencees during grazing by herbivorous zooplankton such as cladocera.

$$\frac{dchla}{dt} = chla \cdot 0.189 \cdot \frac{ps}{ortp + 4.7E-10} \cdot \frac{no3}{no3 + 1.84E-5} \cdot \frac{silica}{silica + 0.06} \cdot \frac{temp}{17.6 - 4.2} \cdot \frac{light}{light + 82.4} - chla \cdot 0.07 \cdot \frac{temp - 5}{16.6 - 3.9} -$$
$$- chla \cdot \frac{0.08}{5} \cdot \frac{temp}{5.1} - clad \cdot 0.2 \cdot \frac{temp}{5.7} \cdot \frac{chla}{chla + 0} \cdot chla \cdot 0.05 \qquad (20.12)$$

$$\frac{dchla}{dt} = chla \cdot 0.042 \cdot \frac{ortp^2}{ortp^2 + 6.2E-6} \cdot \frac{no3^2}{no3^2 + 1.9E-6} \cdot \frac{silica^2}{silica^2 + 0.016} \cdot \frac{temp}{5.8} \cdot light \cdot e^{\frac{\left(\frac{light}{100}+1\right)}{102}} - chla \cdot 0.01 \cdot 1.11^{(temp-17.9)} -$$

$$-chla \cdot 0.025 \frac{temp}{10.8} - chla \cdot \frac{0.04}{5} - clad \cdot 11.6 \cdot \frac{temp}{1} \cdot \frac{chla^2}{chla^2 + 0.9} \cdot chla \cdot 0.02 \tag{20.13}$$

$$\frac{dchla}{dt} = chla \cdot 0.21 \cdot \frac{ortp}{ortp + 0} \cdot \frac{no3^2}{no3^2 + 4E-7} \cdot \frac{silica}{silica + 0.08} \cdot \frac{temp - 4.8}{20 - 0} \cdot \frac{light}{light + 15.3} - chla \cdot 0.038 \cdot 1.11^{(temp-19.5)} - chla \cdot 0.005 \cdot \frac{temp}{5.5} -$$

$$-chla \cdot \frac{0.095}{5} \cdot 1.11^{(temp-17)} - clad \cdot 0.47 \cdot \frac{temp - 0}{15 - 5} \cdot \frac{chla}{chla + 0} \cdot chla \cdot 0.23 \tag{20.14}$$

$$\frac{dchla}{dt} = chla \cdot 0.17 \cdot \frac{ortp}{ortp + 1.2E-9} \cdot \frac{no3^2}{no3^2 + 2.1E-8} \cdot \frac{silica}{silica + 0.63} \cdot \frac{temp}{14.2} \cdot \frac{light}{light + 3.8} - chla \cdot 0.046 \cdot 1.13^{(temp-15)} -$$

$$chla \cdot \frac{0.32}{5} - clad \cdot 0.11 \cdot \frac{temp}{2.6} \cdot \frac{chla}{chla + 0.0005} \cdot chla \cdot 0.13 \tag{20.15}$$

20.4. Conclusions

The software LAGRAMGE for computational assemblage and adaptation of ODE by using the expert knowledge and measured data has been applied for the simulation of chl-*a* in Lake Kasumigaura. As a result two types of chl-*a* models were discovered: (1) chl-*a* equations without considering zooplankton grazing assembled and trained by data of consecutive years were data of the last year was used for testing, and (2) chl-*a* equations considering zooplankton grazing assembled and trained by data of the years 1986 to 1989. The test results of the different models have demonstrated that LAGRAMGE can discover ODE that allow to simulate chl-*a* in Lake Kasumigaura for a variety of years. However the generalisation of discovered equations for unseen data of consecutive years was unsatisfactory, and the accuracy of calculated trajectories with regards to timing and magnitudes of peak events was moderate. The results have highlighted the importance of nutrients as growth limiting factors, and the need for considering functional algae groups in order to appropriately represent their selective grazing by zooplankton.

References

Atanasova N (2005) Priprava in uporaba ekspertnega predznanja za avtomatizirano modeliranje vodnig ekosistemov. PhD Thesis, University of Ljubljana, Ljubljana, Slovenia

Benndorf J, Recknagel F (1982) Problems of application of the ecological model SALMO to lakes and reservoirs having various trophic states. Ecological Modelling 17, 129-145

Bobbin J, Recknagel F (2003) Evolving rules for the prediction and explanation of blue-green algal succession in lakes by evolutionary computation. In: Recknagel F (ed.) (2003) Ecological Informatics. Understanding Ecology by Biologically-Inspired Computation. Springer-Verlag Berlin, Heidelberg, New York, 291-310

Chapra SC (1997) Surface Water-Quality Modeling: McGraw-Hill

DeAngelis DL (1992) Dynamics of Nutrient Cycling and Food Webs. London: Chapmann & Hall

Dzeroski S, Todorovski L (2003) Learning population dynamics models from data and domain knowledge. Ecological Modelling, 170(2-3): 129-140.

Joergensen SE, Bendoricchio G (2001) Fundamentals of Ecological Modelling, Third Ed. Amsterdam: Elsevier Science Ltd.

Kompare B (1995) The Use of Artificial Intelligence in Ecological Modelling. Ph.D. Thesis, FGG, Ljubljana; Royal Danish School of Pharmacy, Copenhagen, Ljubljana, Copenhagen

Langley P, Sanchez J, Todorovski L, Dzeroski S (2002) Inducing process models from continuous data. Paper presented at the The Nineteenth International Conference on Machine Learning, Sydney Australia.

Recknagel F (1989) Applied Systems Ecology. Approach and Case Studies in Aquatic Ecology. Akademie-Verlag, Berlin, 1-138

Recknagel F, Bobbin J, Whigham P, Wilson H (2002) Comparative application of artificial neural networks and genetic algorithms for multivariate time-series modelling of algal blooms in freshwater lakes. Journal of Hydroinformatics 4, 2, 125-134

Recknagel F, Fukushima T, Hanazato T, Takamura N, Wilson H (1998) Modelling and prediction of phyto- and zooplankton dynamics in Lake Kasumigaura by artificial neural networks. Lakes & Reservoirs 3,123-133

Recknagel F (1997) ANNA - Artificial Neural Network model predicting species abundance and succession of blue-green Algae. Hydrobiologia, 349, 47-57

Recknagel F, French M, Harkonen P, Yabunaka K (1997) Artificial neural network approach for modelling and prediction of algal blooms. Ecological Modelling 96, 1-3, 11-28

Recknagel F,. Benndorf J (1982) Validation of the ecological simulation model SALMO. Int. Revue ges. Hydrobiol. 67, 1, 113-125

Straskraba M, Gnauck A (1985) Freshwater Ecosystems, Modelling and Simulation. Elsevier, Amsterdam

Todorovski L (2003) Using Domain Knowledge for Automated Modeling of Dynamic Systems with Equation Discovery. PhD Thesis, University of Ljubljana, Ljubljana, Slovenia

Todorovski L, Džeroski S (1997) Declarative bias in equation discovery. Paper presented at the 14th International Conference on Machine Learning, San Mateo, CA

Vollenweider RA (1968) The scientific basis of lake and stream eutrophication with particular reference to phosphorus and nitrogen as eutrophication factors. Paris: Organisation for Economic Cooperation and Development.

Wei B, Sugiura N, Maekawa T (2001) Use of artificial neural networks in the prediction of algal blooms. Water Research, 35(8): 2022-2028

Whigham P, Recknagel F (2001) Predicting Chlorophyll-a in Freshwater Lakes by Hybridising Process-Based Models and Genetic Algorithms. Ecol. Modelling 146, 1-3, 243-251

Part V

Classification of Ecological Images at Micro and Macro Scale

Chapter 21

Identification of Marine Microalgae by Neural Network Analysis of Simple Descriptors of Flow Cytometric Pulse Shapes

M.F. Wilkins · L. Boddy · G.B.J. Dubelaar

21.1 Introduction

Phytoplankton play a pivotal role in marine ecosystems - collectively fuelling the food web, sometimes forming nuisance blooms, and implicated in climate control. They are sensitive bioindicators in marine ecosystems. Thus, knowledge of species composition, distribution and abundance in the worlds oceans is essential. Traditionally such data have been obtained by microscopic analysis in the laboratory, but this is laborious and time-consuming, abundance estimates are uncertain due to limitations on the number of cells that can be counted, and analysis is often performed a long time after sampling. Analytical flow cytometry (AFC) is a valuable research tool in marine science (Burkill and Mantoura 1990; Jonker et al. 1995) that negates many of these problems. AFC measures various light scatter, diffraction and fluorescence parameters on individual cells, at rates of about 10^3 cells sec^{-1}, providing signatures which can allow taxa to be discriminated.

The vast quantities of non-normally distributed, multivariate data that AFC generates are difficult to analyse by multivariate statistical methods, but artificial neural networks (ANNs; Lippmann 1987; Hush and Horne 1993; Fu 1994; Haykin 1994) have been successfully employed. These typically consist of a three-layered structure of simple data processing elements or nodes (corresponding to the neural cells of their biological counterparts) connected by weighted connections. The input layer contains one node for each input parameter, while the output layer contains one node corresponding to each of the potential categories to which the input pattern may belong. The network is trained to recognise the different categories of data via a learning procedure, during which the network is repeatedly presented with labelled examples of each data category and the internal weighted connections between nodes are modified to produce a network output that more nearly reflects the correct identification. Following training an unknown data pattern can be presented to the network and the network output indicates the most

likely category for the data pattern. ANNs have the advantage in comparison to multivariate statistics of making no prior assumptions as to the nature of the category data distributions, and once trained, they are fast and efficient in use. In most of the early studies applying ANNs to AFC data only a few taxonomic categories were discriminated (e.g. Frankel et al. 1989, 1996; Morris et al. 1992; Balfoort et al. 1992; Smits et al. 1992; Wilkins et al. 1994, 1996). Scaling up is not a trivial task, but 36 to 72 phytoplankton species, grown in artificial culture, have now been discriminated (with 70% overall successful identification) using multilayer perceptron (MLP) and radial basis function (RBF) ANNs (Boddy et al. 1994, 2000; Wilkins et al. 1999). The latter offers significant advantages in the ability to reject data patterns not corresponding to any of the classifications known to the network (Morris and Boddy 1996, Wilkins et al. 1999). While some species are always identified with high success, others are not, due to overlap of character distributions. To improve discrimination, additional and/or different discriminatory characters are required.

Conventional flow cytometers use analog data capture electronics to collect summary statistics for each pulse, such as pulse width, peak pulse height and integrated value. However, the pulse shape of the light scatter and fluorescence signals, acquired as the analysed particle traverses the beam focus, may well contain additional discriminatory information which can be exploited (Godavarti et al. 1996). CytoBuoy, an autonomous AFC designed for mounting within a buoy for *in situ* sampling (Dubelaar et al. 1999; Dubelaar and Gerritzen 2000) uses a single green (532 nm) laser and retains full digital pulse shape information for four signals: forward scatter (FSC), side scatter (SSC), orange fluorescence (FLO) and red fluorescence (FLR), sampled at 4MHz. The raw 8-bit sample values for each signal are accumulated in an internal 64kB data buffer before transmission to the shore by radio link; this limits the number of particles in any one sampling run to a few thousand, depending on particle size.

The CytoBuoy is designed to process a relatively wide sample stream. To maintain a sufficiently large depth of focus over this stream, the laser focus width cannot be reduced to less than 5µm. The measured pulse shapes are the convolution of the light intensity profile across the laser focus with the particle shape (or distribution of optically emitting material) along its longest axis (as particles flowing through the laser beam are stretched by the fluid acceleration). The measured pulse shapes of particles smaller than about 5µm are essentially dominated by the Gaussian shaped light intensity profile, with little or no influence from the particle shape itself. The shape features of particles bigger/longer than a few times the laser focus width (about 20 µm and larger) are well expressed in the detector pulses (Fig. 21.1a), whereas for intermediate sized particles (roughly 5 – 20 µm) the shape expression varies from little to reasonable.

In this paper neural net analysis of CytoBuoy pulse shape data is investigated. In order to apply ANNs to pulse shape analysis, the first step is that of feature extraction: converting the raw pulse representation (i.e. a variable length sequence of sample values) into a more compact representation capable of capturing the variation between the different categories (i.e. a small number of characteristic measurements, each quantifying a different aspect of the pulse shape). In addition

it is desirable, if possible, to separate out information on particle size and overall maximum pulse intensity from the shape information, to investigate the discriminability of different categories from pulse shape alone; the extra information can then be supplied to the network via additional input parameters.

The following biological questions are specifically addressed: does pulse shape improve discrimination between taxa? Which taxa can easily be discriminated using pulse shape information alone and which cannot? Which taxa form clear groups under pulse shape? If any of the taxa that can be identified on pulse shape alone are taxa that are difficult to identify by the summary statistics usually used, what features of the pulse shape are important in discrimination, and with what structural/morphological features are they correlated?

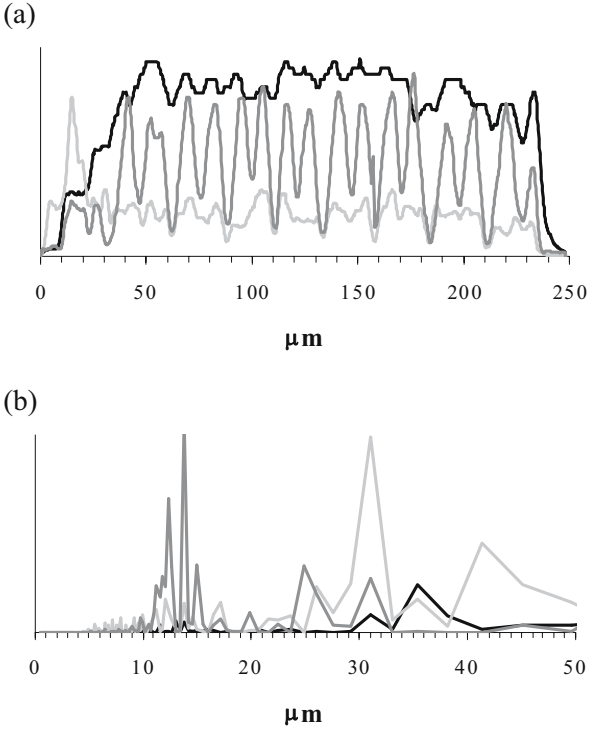


Fig. 21.1. (a) CytoBuoy pulse shapes for FSC (black trace), SSC (light grey trace), FLR (dark gray trace) for an example of an unknown chain-forming diatom from a natural sample (the FLO signal is negligible and not visible). The cells form doublets, clearly seen in the FLR and SSC traces. The total chain length is 248 µm. (b) the corresponding DCT power spectra; the FLR trace has a peak at 13.7 µm, corresponding to the spacing between the cells of the doublets, while the SSC trace has a peak at 31 µm, corresponding to the spacing.between doublets.

Table 21.1
The performance of RBF networks trained on pulse shape parameters alone (6 parameters), summary statistics alone (12 parameters) , and combined pulse shape parameters and summary statistics (18 parameters) tested on an independent data set of 500 data patterns per species (for details of parameters see text). Figures are the percentage of test data correctly identified with estimated 95% confidence limits (5 replicates)

	Approx.size (μm)	Pulse shape parameters		Summary statistics		Combined	
Amphidinium carterae	15-20	30	± 2	67	± 4	73	± 3
Aureodinium pigmentosum	7-12	17	± 2	92	± 1	93	± 3
Chaetoceros simplex var. *calcitrans*	4-6	29	± 2	68	± 4	68	± 10
Chroomonas salina	5-12	38	± 5	93	± 3	92	± 2
Chrysochromulina camella	6-12	22	± 6	49	± 10	50	± 4
Chrysochromulina chiton	5-9	19	± 1	65	± 4	69	± 2
Chrysochromulina polylepis	6-8	15	± 4	40	± 7	41	± 3
Cryptomonas calceiformis	10-15	77	± 3	80	± 3	82	± 4
Cryptomonas maculata	12-20	51	± 4	84	± 2	92	± 2
Cryptomonas reticulata	18-25	54	± 3	71	± 4	79	± 5
Dunaliella minuta	3-12	20	± 1	35	± 8	43	± 11
Dunaliella tertiolecta	6-12	45	± 4	74	± 3	79	± 1
Emiliana huxleyi	5-7	27	± 2	64	± 2	65	± 5
Gymnodinium simplex	6-10	4	± 2	69	± 3	69	± 3
Hemiselmis brunnescens	5-8	51	± 5	89	± 2	86	± 2
Hemiselmis virescens	5-8	45	± 3	98	± 0	97	± 1
Imantonia sp.	3-8	35	± 1	69	± 5	68	± 5
Isochrysis galbana	4-8	22	± 2	82	± 3	83	± 3
Ochrosphaera neopolitana	8-10	28	± 3	88	± 2	91	± 2
Pavlova lutheri	4-6	17	± 2	75	± 2	75	± 2
Phaeocystis pouchetii	3-6	13	± 7	71	± 4	70	± 4
Plagioselmis punctata	6-9	21	± 5	91	± 3	93	± 2
Platychrysis sp.	6-12	5	± 2	22	± 11	13	± 7
Prorocentrum minimum	16-18	5	± 2	54	± 6	54	± 11
Prorocentrum nanum	8-10	6	± 2	26	± 13	27	± 6
Prymnesium parvum	8-10	7	± 2	33	± 9	39	± 9
Pyramimonas grossii	5-10	22	± 3	56	± 7	53	± 4
Pyramimonas obovata	4-8	1	± 1	57	± 9	59	± 9
Rhinomonas salina	*5-10*	49	± 3	93	± 2	94	± 1
Rhodomonas sp.	8-13	43	± 2	90	± 1	91	± 2
Synechococcus sp.	1-3	66	± 6	97	± 2	96	± 1
Tetraselmis striata	6-8	35	± 4	37	± 3	49	± 5
Tetraselmis suecica	6-15	9	± 2	24	± 14	31	± 18

21.2
Materials and Methods

21.2.1
Pulse Shape Extraction

CytoBuoy pulse shape data for 36 small unicellular phytoplankton species in pure culture (Table 21.1) were collected during the course of the AIMS project (CEC grant no. MAS3-CT97-0080). These species were selected from a much larger database of species as the only Each species was represented by four data files containing 64kB of 8-bit sample values. Pulse shape data for 1000 particles per species were extracted from these files by purpose-written software. The length of each pulse, log peak value and log mean value were found, and particles for which the FSC pulse was less than four sample values in length (2μm) were rejected. The pulses were normalised using a cubic spline interpolation procedure (Press et al. 1992) to a standard number of samples and to unit integral value. This normalisation procedure effectively separates out information describing pulse shape from global pulse characteristics such as length and area, allowing independent assessment of the contribution of each type of information to species discrimination. The number of samples used to represent each pulse was chosen to be a power of two plus one, for subsequent compatibility with the frequency-space decomposition methods used; 33 samples was found to be an adequate description of the pulse shapes for the species in this study, although other species may require more (e.g. chain-formers; Fig. 21.1a).

21.2.2
Data Filtering

The data for each species were plotted and pulses corresponding to particles with low integral red fluorescence (a measure of chlorophyll content) were removed; such particles are frequently present in phytoplankton cultures and typically represent cellular debris, dead cells or bacterial contamination of the culture.

21.2.3
Data Transformation

A discrete cosine transformation (DCT) procedure (Press et al. 1992) was used to decompose all pulses into a frequency space representation, expressing the original pulse as the superposition (sum) of a number of separate components taking the form of cosine waves with differing amplitude and frequency. The reason for this is twofold. Such a representation effectively separates out elements

of structure due to the presence of features with different length scales in the pulse. The representation is also more compact. Typically only a few of the low-frequency components have any significant amplitude, with the amplitude of the remaining high-frequency components being close to zero. Thus, instead of using all 33 sample values to represent the pulse, the overall shape of the pulse can be approximated with reasonable accuracy using relatively few cosine coefficients. The square of the amplitude coefficients represents the power spectrum of the pulse (the amount of energy contained in components of each frequency). The position of the peak of the power spectrum can be used to define a "characteristic length" for each pulse- this is the length scale of the most significant features detected by the pulse. For example, a chain-forming diatom gives rise to a signal with regularly-spaced peaks (Fig. 21.1a), leading to a peak in the power spectrum at the spatial frequency corresponding to the length of the individual cells in the chain (Fig. 21.1b).

21.2.4 Principal Component Analysis

Principal component analysis (Joliffe 1986) was used to investigate the extent to which the variation between the normalised pulse shapes seen in the data set could be explained by the combination of a small number of independent modes of variation. The mean of the DCT transformed data was estimated and used to generate a "mean pulse shape" for each parameter, the average of all the normalised pulses over the data for all 36 species. The effect of variation from the mean pulse shape along each of the first four principal component axes was plotted (Fig. 21.2). The first mode was characterised primarily by a shift in the position of the FLO peak (corresponding to the presence of phycoerythrin) with respect to the peaks for the other signals. The second mode consisted of simultaneous sharpening of the SSC and FLO peaks accompanied by the development of a double peak in the FSC signal. The third mode consisted of a shift in the position of the SSC peak, while the fourth mode was similar to the second except that sharpening of the FLO peak was accompanied by flattening of the SSC peak and vice versa. Higher modes showed more complex shape changes. Several species showed pronounced bimodal distributions in the first mode of variation, e.g. *Cryptomonas maculata* and *Rhinomonas salina* – this may be caused not by the presence of two distinct subpopulations but rather be due to cells with inherent asymmetry passing through the instrument in opposite orientations. Supporting evidence for this conclusion comes from examining the between-class variance as a percentage of the explained variance for each mode (Table 21.2), showing that while the first mode accounts for the largest fraction (21.2%) of the total variance, only 2.4% of this is due to between-class variance: there is thus very little discriminatory information in this mode, implying that the cause of this variation is common to cells of all species.

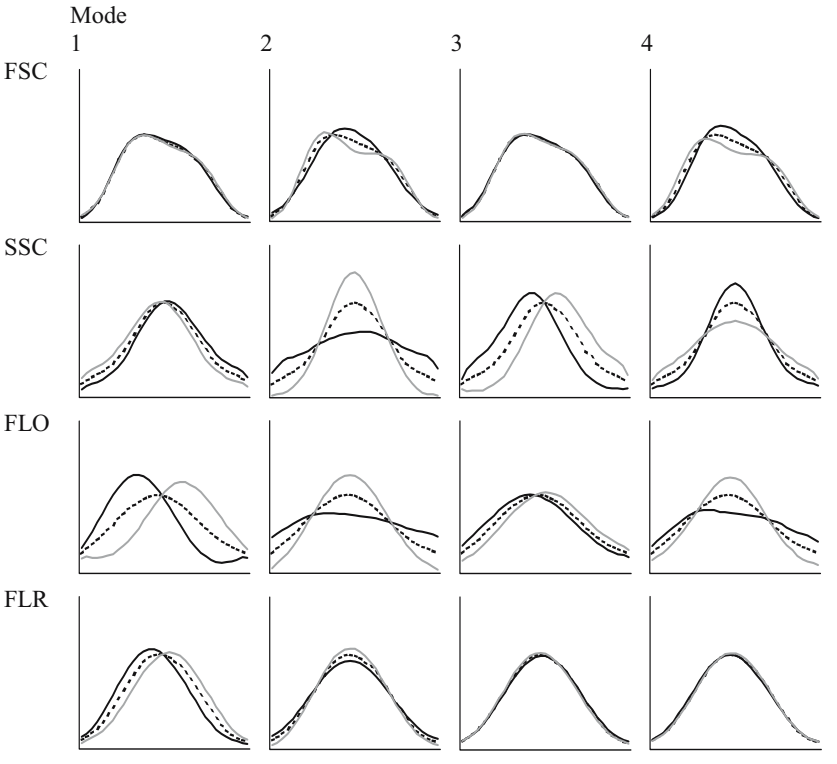


Fig. 21.2. The first four principal modes of variation of normalised pulse shapes, together with the tenth mode, from DCT data set (1000 patterns from each of 36 species). The black and grey lines show one standard deviation positive and negative deviation respectively from the mean pulse shape (dotted line). Modes 1 and 3 are characterised primarily by shifts in the position of the FLO and SSC peaks respectively, while modes 2 and 4 are characterised primarily by simultaneous variation in the sharpness of the SSC and FLO peaks. The amount of variance explained by each mode is given in Table 21.2.

Table 21.2
The explained variance (as a percentage of the total variance) and the between -class variance (as a percentage of the explained variance) for each of the first twelve modes of variation.

Mode	Explained variance (% of total)	Between-class variance (% of explained variance)
1	21.2	2.4
2	16.0	57.8
3	13.4	3.6
4	8.0	42.2
5	5.2	3.3
6	4.6	36.8
7	3.0	0.6
8	2.8	4.4
9	2.3	3.8
10	2.2	11.7
11	1.8	22.3
12	1.5	11.1

21.2.5 Neural Network Analysis

The available data was partitioned into independent training and test data sets, both containing data for 500 particles per species. Radial basis function neural networks were trained to discriminate the 36 species using (i) pulse shape information only; (ii) summary statistics only; (iii) combined pulse shape and summary statistics. The pulse shape information for each particle consisted of six parameters, the six principal components of the DCT-transformed training data found previously to contain the most discriminatory information (components 2, 4, 6, 10, 11, 12). The summary statistic information for each particle consisted of twelve parameters, the pulse length, peak pulse value, and mean pulse value for each of the four signals (FSC, SSC, FLO and FLR). The combined pulse shape and summary statistics thus had a total of eighteen parameters.

All RBF HLNs used the Mahalanobis distance metric, implementing multivariate Gaussian basis functions, and were initially positioned using the Kohonen LVQ algorithm (Kohonen 1990); networks of this type have previously been shown to perform well (Wilkins et al. 1999). The number of HLNs was

automatically determined using the orthogonal least squares learning algorithm to select an optimal subset from a large pool of candidate HLNs (Chen et al. 1991).

Following training, the test data set was used to assess the network performance in terms of proportion of correct identification, and to determine which species were consistently misidentified. Each network was replicated five times and the results averaged.

21.2.6 Hardware and Software

All RBF networks were simulated in software using *AimsNet* (http://www.cf.ac.uk/biosi/staff/wilkins/aimsnet), a software package developed by one of the authors (M.F.W.) under the AIMS project (CEC grant no. MAS3-CT97-0080) running on a 500MHz PC under WindowsNT.

21.3 Results

Using the pulse shape data alone, six species (*Synechococcus* sp., *Cryptomonas calceiformis*, *Cryptomonas maculata*, *Cryptomonas reticulata*, *Tetraselmis tetrathele* and *Hemiselmis brunnescens*) were all discriminated correctly with greater than 50% accuracy; however over the whole test data set only 28.6% of data patterns were correctly recognised (Table 21.1). The corresponding figure using summary statistics alone was 67.2%; this rose slightly to 69.2% when the pulse shape and summary statistic information were combined. The addition of pulse shape information generally improved the discrimination between species from the same genus, e.g. *Tetraselmis* (4 species), *Chrysochromulina* (3 species), *Cryptomonas* (3 species) and *Dunaliella* (2 species). For example, the amount of *T. striata* misidentified as *T. tetrathele* was reduced by a factor of two; plotting the pulse shapes for these species shows that while the SSC, FLO and FLR signals are similar, the FSC signal for *T. tetrathele* shows a pronounced asymmetry which can be exploited to aid discrimination (Fig. 21.3). For the three *Cryptomonas* species, the FSC signal for *C. maculata* shows two distinct peaks that are significantly closer together than for the other two species (Fig. 21.4).

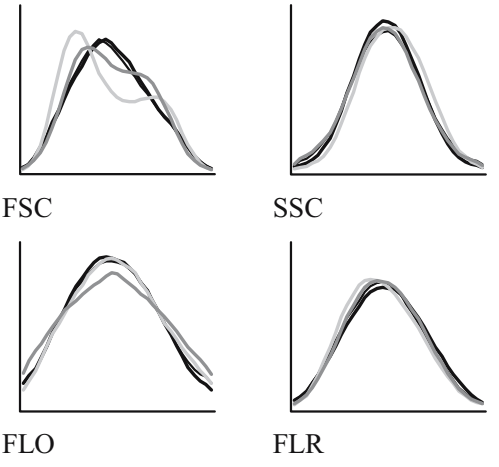


Fig. 21.3. The normalised mean pulse shapes for the four *Tetraselmis* species, showing the distinctive asymmetry of the FSC signal for *T. tetrathele*. Key: *T. striata* (thick black line); *T. suecica* (thin black line); *T. tetrathele* (light grey line); *T. verrucosa* (dark grey line).

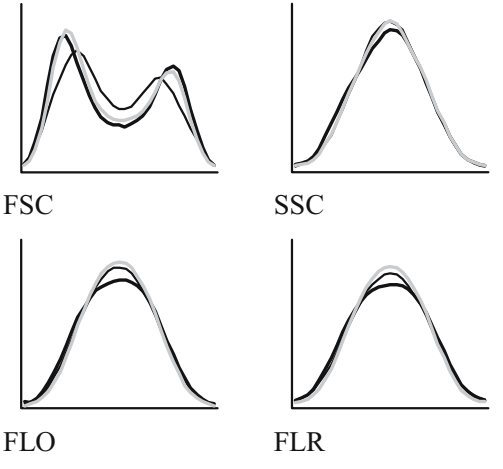


Fig. 21.4. The normalised mean pulse shapes for the three *Cryptomonas* species, showing the smaller interpeak separation in the FSC signal of *C. maculata* in comparison to the other two species. Key: *C. reticulata* (light grey line); *C. maculata* (thin black line); *C. calceiformis* (thick black line).

21.4 Discussion

These results demonstrate that the incorporation of pulse shape information provides a modest increase in discriminating power, and that principal component analysis can be used to separate out genuine variation between pulse shapes, due to differences between the organisms under study, from spurious variation arising from the physics of the data capture process.

The data used in this study all originated from small single-celled organisms, a consequence of the difficulty of obtaining sufficient training data for larger organisms due to the limited internal buffer space of the CytoBuoy. The pulse shape information for larger organisms, e.g. colony and chain forming species, contains proportionately more discriminatory information (Fig. 21.1). Such particles can be up to a millimeter in length (2000 samples).

The recognition performance of 69.2% for 36 species appears low by comparison with previous studies: for example 72 taxa were identified with 70% success from FACSort data (Boddy et al., 2000), while 34 species were discriminated with over 90% correct recognition from EurOPA data (Wilkins et al., 1999). This latter result is due to the higher resolution of the EurOPA cytometer, which also incorporated two-laser excitation and additional diffraction parameters as an index of particle shape. The CytoBuoy prototype has a resolution 2-3 times lower than the EurOPA, due to compromises in the optics imposed by its compact design and autonomous mode of operation. The optics of the CytoBuoy have since been substantially redesigned in order to improve sensitivity and resolution.

21.5 Conclusions

The use of AFC pulse shape information does improve discrimination of microalgal taxa, and is likely to be even more useful when species that form chains are to be discriminated. The use of RBF ANNs was again shown to be a rapid and useful tool for analysing large sets of high dimensional data.

Acknowledgements

This research, together with the *AimsNet* software, forms part of AIMS, a project funded by the Commission of the European Community, CEC grant no. MAS3-CT97-0080. We thank all partners for valuable discussion.

References

Balfoort HW, Snoek J, Smits JRM, Breedveld LW, Hofstraat JW, Ringelberg J (1992) Automatic identification of algae: neural network analysis of flow cytometric data. J. Plankton Res. 14, 575-589

Boddy L, Morris CW, Wilkins MF, Tarran GA, Burkill PH (1994) Neural network analysis of flow cytometric data for 40 marine phytoplankton species. Cytometry 15, 283-293

Boddy L, Morris CW, Wilkins MF, Al-Haddad L, Tarran GA, Jonker RR, Burkill PH (2000) Identification of 72 phytoplankton species by radial basis function neural network analysis of flow cytometric data. Mar. Ecol. Prog. Ser. 195, 47-59

Burkill PH, Mantoura RFC (1990) The rapid analysis of single marine cells by flow cytometry. Phil. Trans. Royal Soc. A333, 99-112

Chen S, Cowan CFN, Grant PM (1991) Orthogonal least-squares learning algorithm for radial basis function neural networks. IEEE Trans. Neural Networks 2, 302-309

Dubelaar GBJ, Gerritzen PL, Beeker AER, Jonker RR, Tangen K (1999) Design and first results of CytoBuoy: a wireless flow cytometer for in situ analysis of marine and fresh waters. Cytometry 137, 247-254

Dubelaar GBJ, Gerritzen PL (2000) CytoBuoy: a step forward towards using flow cytometry in Operational Oceanography. Scientia Marina (in press)

Frankel DS, Olson RJ, Frankel SL, Chisholm SW (1989) Use of a neural net computer system for analysis of flow cytometric data of phytoplankton populations. Cytometry 10, 540-550

Frankel DS, Frankel SL, Binder BJ, Vogt RF (1996) Application of neural networks to flow cytometry data analysis and real-time cell classification. Cytometry 23, 290-302

Fu LM (1994) Neural Networks in Computer Intelligence. McGraw-Hill, New York

Godavarti M, Rodriguez JJ, Yopp TA, Lambert GM, Galbraith DW (1996) Automated particle classification based on digital acquisition and analysis of flow cytometric pulse waveforms. Cytometry 24, 330-339

Haykin S (1994) Neural networks: a comprehensive foundation. Maxwell Macmillan International, New York

Hush DR, Horne BG (1993) Progress in supervised neural networks- what's new since Lippmann? IEEE Sig. Proc. Mag. 10, 8-39

Joliffe IT (1986) Principal components analysis. Springer-Verlag, New York

Jonker RR, Meulemans JT, Dubelaar GBJ, Wilkins MF, Ringelberg J (1995) Flow cytometry: a powerful tool in analysis of biomass distributions in phytoplankton. Water Sci. Technol. 32, 177-182

Lippmann RP (1987) An introduction to computing with neural nets. IEEE Acoustics, Speech and Signal Proc. Mag. 4, 4-22

Morris CW, Boddy L, Allman R (1992) Identification of basidiomycete spores by neural network analysis of flow cytometry data. Mycol. Res. 96, 697-701

Morris CW, Boddy L (1996) Classification as unknown by RBF networks: discriminating phytoplankton taxa from flow cytometry data. In: Dagli, C.H., Akay, M.,Chen, C.L.P., Fernandez, B.R., Ghosh, J. (Editors), Intelligent Engineering Systems Through Artificial Neural Networks, vol. 6. ASME Press, New York, pp 629-634

Press WH, Flannery BP, Teukolsky SA, Vetterling WT (1992) Numerical recipies in C: the art of scientific computing. Cambridge University Press, 994 pp

Smits JRM, Breedveld LW, Derksen MWJ, Kateman G, Balfoort HW, Snoek J, Hofstraat JW (1992) Pattern classification with artificial neural networks: classification of algae, based upon flow cytometer data. Anal. Chim Acta 258, 11-25

Wilkins MF, Boddy L, Morris CW, Jonker R (1996) A comparison of some neural and non-neural methods for identification of phytoplankton from flow cytometry data. CABIOS 12, 9-18

Wilkins MF, Morris CW, Boddy L (1994) A comparison of radial basis function and backpropagation neural networks for identification of marine phytoplankton from multivariate flow cytometry data. CABIOS 10, 285-294

Wilkins MF, Boddy L, Morris CW, Jonker RR (1999) Identification of phytoplankton from flow cytometry data using radial basis function neural networks. Appl. Env. Microbiol. 65, 4404-4410

Chapter 22

Age Estimation of Fish Using a Probabilistic Neural Network

S.G. Robertson · A.K. Morison

22.1 Introduction

Age composition data provide fundamental insights into fish biology and stock productivity and allow the estimation of the basic parameters for describing growth, mortality rates and recruitment. Much time and money is spent on the collection and preparation of samples, and skilled technicians labour for many hours at microscopes, counting increments in the prepared structures. It is estimated that over 1 million fish were aged worldwide in 1999, mostly using scales and otoliths (Campana and Thorrold 2001). However, the process is somewhat subjective and there is much interest in automating the process and making estimates more reliable. To date none of the tested methods have been successful. A pilot study by Robertson and Morison (1999) first suggested that neural networks may provide the way forward for this previously intractable problem. In this paper we firstly give a brief account of traditional approaches to age estimation. We then describe the previous attempts to develop automatic or computer-aided methods and the problems they have encountered. Finally we describe the results of a recent application of a probabilistic neural network to the process of age estimation in fish and discuss the strengths of this novel approach.

22.2 Traditional Methods of Age Estimation

Fisheries scientists have been estimating the age and growth of fish by various means since it was first realised that modes in length-frequency distributions could be used to estimate the age of discrete cohorts. Since then methods of estimating fish age have proliferated but usually involve counting of growth increments on hard parts (scales, vertebrae, fin rays, fin spines, eye lenses and otoliths). On otoliths (usually the structure of choice for all but the youngest of fish), these

increments are composed of alternating opaque and translucent zones. The technique is analogous to the counting of tree rings. More recently, radiometric methods (Bennett *et al.* 1982) and trends in radiocarbon levels (Kalish 1993) have provided independent estimates of age. However, these methods are even more expensive and time consuming and are mainly used to validate (show the accuracy of) ages estimated from increment counts. The counting of growth increments on otoliths is still the most commonly employed method of age estimation.

The process of age estimation by counting growth increments is not straightforward and requires considerable experience to be done reliably. Age estimates may be made as part of a validation study, for descriptions of age composition and life history parameters, or as part of ongoing or "production" ageing for commercially exploited species (Morison *et al.* 1998a). This ongoing process providing estimates of the age composition of the catch from commercial fisheries over many years also requires special attention to the issue of maintaining accuracy across years, and often across readers. The importance of the ageing data for fisheries assessments, and the recognition of problems of consistency in age estimates, have led to considerable attention being given to the procedures for the prevention, identification and quantification of biases and errors in age estimates (Beamish and Fournier 1981; Chang 1982; Kimura and Lyons 1991; Richards *et al.* 1992; Campana *et al.* 1995; Morison *et al.* 1998a; Gröger 1999; Campana 2001). The problem of validation of a method of estimating age is a important issue in itself that is often poorly done. This was first highlighted by Beamish and McFarlane (1983) and has been recently revisited by Campana (2001).

Differences between laboratories and between readers are common, and are not helped by differences in methods of sample preparation. The species and the structure largely determine the preferred preparation techniques, but there is also strong conservative element as particular laboratories will favour particular preparation methods. A conservative approach is due partly to the available equipment and the skills of staff, partly to the need to maintain consistency over years, but also to the familiarity of staff with interpreting a particular form of preparation. Staff become used to viewing specimens in a particular orientation and form of illumination (using either transmitted or reflected light) to the extent that they can find it difficult to interpret samples presented differently. This is similar to the difficultly people have recognising faces presented upside down or as a negative image, and points to an important aspect of the process: it is as much one of pattern recognition as of counting increments. However, as a pattern recognition problem it is probably closer to the character recognition in hand writing than face recognition as there are a limited number of groups (age classes) to be recognised. There may be much individual variation but each age class shares common features.

22.3
Approaches to Automation in Fish Age Estimation

Williams and Bedford (1974) remarked that "otolith reading remains ... as much an art as a science" and the subjective nature of the interpretation of the increments has driven the search for more objective methods. In some instances these have exploited the correlation between otolith growth and fish by developing linear regressions relating fish age and various measures of otolith size or weight (Boehlert 1985; Fletcher 1995). The statistical approach using correlates of fish age has not been widely adopted. Its use for the age estimation of short-lived species such as pilchards (*Sardinops neopilchardus*) (Fletcher 1995) owes as much to the great difficulties experienced with the interpretation of the highly variable increment patterns on the otoliths of these species, as to the efficacy of the technique. Spatial and temporal variation in these relationships also create problems for the routine application of such methods (Worthington *et al.* 1995).

The processing power of modern computers and image analysis software has stimulated much interest in their ability to automate the increment recognition and age estimation process. At one level these have required or allowed interaction with the user to assist in the identification of increments (Macy 1995; Cailliet 1996). More advanced methods have sought algorithms that identified increments as minima and maxima in digital profiles of brightness level (Troadec 1991; Welleman and Storbeck 1995; Lagardère and Troadec 1997; Morison and Robertson 1997; Takashima *et al.* 2000).

One problem with these approaches is that they assume that one growth increment is formed each year. This is usually true, but it is also common that such increments are comprised of two or more distinct sub-annual increments, called false increments. The ability to consistently distinguish annually formed increments is the valuable skill that an experienced reader brings to the process. Recognition of peaks and troughs in a brightness profile is statistically simple, but discerning which should be counted and which excluded is not. These methods are also only reliable on samples with very clear increments, but such samples are usually in the minority.

We have tried the 'find-the-peaks-and-count-them' approach for these species in a pilot study comparing different automation methods and found them to be satisfactory only in a very narrow range of circumstances (Morison and Robertson 1997). The signals are inherently noisy and the change in slope method for identifying peaks (e.g. Takashima *et al.* 2000) requires specifying a particular search bandwidth. As increments usually become narrower towards the otolith margins, this bandwidth must be progressively reduced to work equally reliably on the broad inner increments, when growth is fast, and on the narrow outer increments, formed when growth slows. Signal processing techniques have also been applied to these data involving scaling, demodulation and spectral analysis, to extract the number of cycles as an age estimate (Troadec 1991; Lagardère and Troadec 1997; Troadec *et al.* 2000). However, in either case the approach requires a rigid algorithm that reflects the expected reduction in increment spacing. The

expected pattern has first to be identified and numerically coded and therefore the methods do not cope well with variations from the standard pattern.

Neural networks may side-step this problem. We showed that a back-propagation neural network, using single transects across otolith images, had the potential to reproduce the age estimates of an experienced reader with acceptable precision for two of the three species tested (Robertson and Morison 1999). The method worked better than any previous methods and for a greater range of age classes.

Using neural networks to automate the fish age estimation is a fundamentally different approach to those used previously. Past attempts have mostly tried to duplicate the outcome by duplicating the process used by human readers: identify each valid annual increment and then count them. Using neural networks aims to duplicate the outcome without identifying the process. This pragmatic approach focuses on the quality of the age estimates without putting effort into developing algorithms to identify increments in images. It recognizes that the process used by human readers is enormously complex. Experienced otolith readers use a combination of features of each otolith to assist them in making age estimates. They consider the shape of the otolith, the width and spacing of increments and how this changes from the otolith centre (the primordium) to the edge, the consistency or completeness of increments around the image, the sharpness of the transition from opaque to translucent zones. Even for fish of the same age these patterns have significant individual variation, and may vary between species, with sex and maturity, with fish size, and among different sub-populations, and from year to year. In interpreting this variation readers employ their experience from having viewed many thousands otolith of the same species. The way this is done is difficult to codify into strict decision rules. This is further complicated in that weight given to different aspects may vary among readers, without necessarily compromising the accuracy of the estimated ages. The results of the initial trial of a neural network were encouraging but the sample size was very small (40 individuals) and the testing method did not use a completely independent sample set to evaluate network performance. We were also aware that other network types and additional data inputs may substantially increase the effectiveness of the networks. A more rigorous test of the approach was needed.

22.4 The Application of a Probabilistic Neural Network to Fish Age Estimation

The same three species were used in these experiments as in the pilot study: *Pagrus auratus* (Sparidae) (987 samples), *Acanthopagrus butcheri* (Sparidae) (913 samples) and *Macruronus novaezelandiae* (Merlucciidae) (1531 samples). Age estimates for these samples had previously been made using the procedures and protocols described by Morison *et al.* (1998a). Thin-sections of sagittal otoliths viewed with transmitted light were used for all species, and the number of

opaque increments used to estimate the age. This method of age estimation has previously been validated for *A. butcheri* by Morison *et al.* (1998b), for *P. auratus* by Francis *et al.* (1992), and for *M. novaezelandiae* by Kalish *et al.* (1997) and Punt *et al.* (2001). The samples used in this study do not come from known-age fish, but the ages were estimated by experienced readers with many years of experience with each of the species.

Images of the otolith sections were saved as eight bit greyscale tagged image format files. From each of the images, luminance data were collected from up to five transects for all species (the signal data). The transects lines were marked with the cursor on the images from the primordium of the otolith to the edge of the otolith. The transects were marked on the areas of the otolith which showed a clear alternating patterns of opaque and translucent zones that would be used by an experienced reader when estimating age. The location and number of transects varied among species but were consistent for all individuals within species (Figures 22.1 to 22.3). Three transects were taken from blue grenadier, five from the other two species. For each transect, its length was calculated (in number of pixels) from the XY coordinates of the start and finish points.

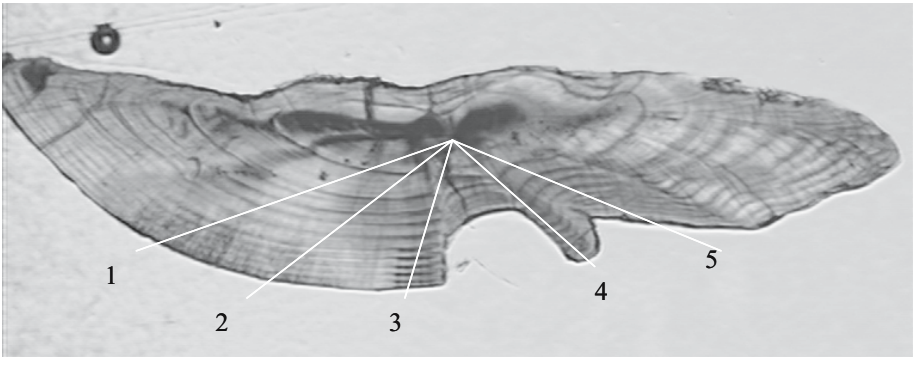


Figure 22.1. Sectioned snapper otolith showing typical locations of transects. Transects were sampled from the ventral lobe (1) then successively counter - clockwise around the otolith from the primordium. The example is of a 10 year old, 60 cm snapper sampled on 5/3/97 from Port Phillip Bay.

Signal data from otolith images was pre-processed with a discrete fast Fourier transform (DFT). As the DFT transform requires that the array length be 2^n pixels, the original signal data were first mapped to an array of 128 values. The harmonic (absolute) values of the first 21 complex numbers were used as inputs to the neural network. Reconstructed profiles from 21 complex numbers had been found to accurately reproduce the originals (Figure 22.4). In addition to the signal data from the otolith images, other data used as inputs to neural networks included fish length (cm), otolith weight (g), sex, area of capture, and date of capture. Sex was expressed as a categorical variable. Date of capture was expressed as a decimal number representing the proportion of the year from 1 January. The number of

input variables used for the neural networks thus totalled 114 for *A. butcheri* and *P. auratus*, and 70 for *M. novaezelandiae*. A summary is given in Table 22.1.

A probabilistic network was used for the test and implemented using proprietary software (Ward Systems Neuroshell®). Probabilistic neural networks (PNNs) are intrinsic classification models and are known for their ability to train quickly (Masters 1993). PNNs categorise data into a specified number of output categories that correspond to, in this application, age-classes. The topology of the PNNs resembles the back propagation neural network i.e. there are three layers in the networks. The difference lies in the number of neurons in the hidden layer and the function of the hidden layer. There are as many neurons in the hidden layer as there are samples in the pattern dataset. The input layer uses the same linearly scaled data as the input layer of the back propagation models. The output layer

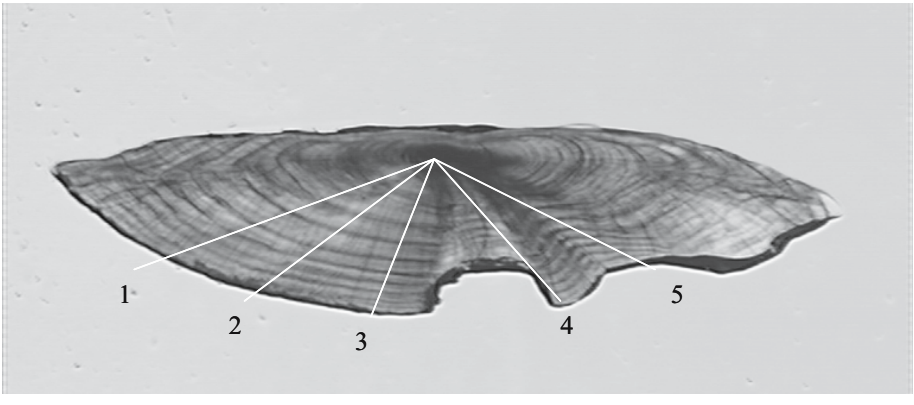


Figure 22.2. Sectioned black bream otolith showing typical locations of transects. Transects were sampled using the same locations and sequence as the transects taken from the snapper otolith sections. The example is of a 9-year-old 25 cm black bream sampled on 1/4/97 from Sydenham Inlet.

has the same number of neurons as the number of age classes. The probabilistic neural network provides a probability density function of age-membership as an output (i.e., all the outputs sum to one) where the most probable age-class is classified by the output neuron with the highest value. The hidden layer in the PNN uses a 'sphere of influence' weighting function (a multi-variate extension of Parzen's method (Masters 1995)) to classify the given inputs to a particular age-class. The width of the 'sphere of influence' is determined by a scaling parameter that varies between input variables. There is no objective method for determining the size of this scaling parameter (Masters 1994). Neuroshell® software uses a 'genetic' algorithm for determining the optimum size of the scaling parameter for each age-class.

Datasets were randomly divided into training, test, and validation subsets in the ratio of 3:1:1. The training set was used for minimising or fitting the model to produce the desired output response for a given data input. The test set was used to

evaluate when the training has reached an optimal level. The presentation of all the training samples to the neural network and genetic optimisation of the scaling parameters marked the completion of one generation. The training was continued until no change in the error term for the test set was observed in 20 generations. The production set provided an estimate on the final performance of the trained model, using a dataset not previously used in the training (minimisation) or test phases of model fitting.

Table 22.1. Summary of input variables used for training neural networks.

Dataset	Variables	Data type
Biological	Otolith weight (g)	Continuous
	Fish length (cm)	Continuous
	Sex	Categorical
	Date of capture	Continuous
	Transect lengths (pixels)*	Continuous
Signal	First 21 harmonics from DFT*	Continuous

*Data were collected from five transects per individual for *A. butcheri* and *P. auratus* and three per individual for *M. novaezelandiae*.

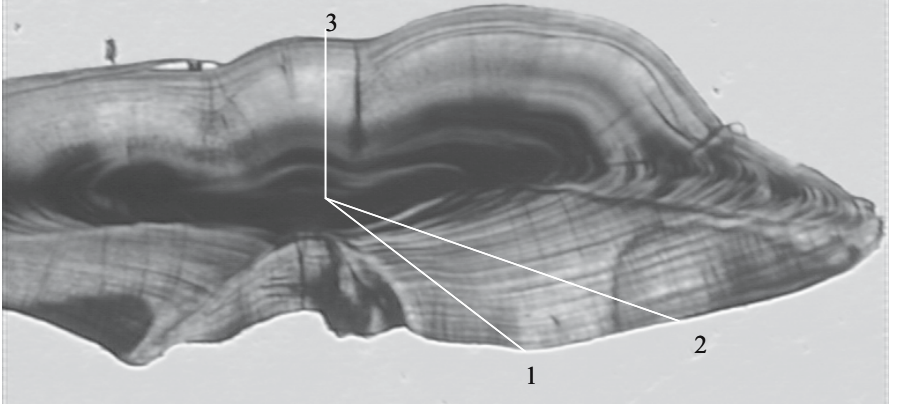


Figure 22.3. Sectioned blue grenadier otolith showing typical locations of transects. The first two transects were drawn from the primordia to the edge of the ventral lobe. The third transect was drawn from the primordium to the distal edge of the otolith. The example is of an 8 year old 93 cm female blue grenadier captured on 29/6/98 from the west coast of Tasmania.

Network performance was evaluated by two criteria. Firstly, an index of average percent error (IAPE) (Beamish and Fournier 1981) was calculated as a measure of precision with success being quantified as a value of less than 10%

(Morison *et al.* 1998a). Secondly, the slope and intercept for a linear regression of predicted on observed ages was used as a measure of bias, with success being quantified as no significant differences from 1 and 0 respectively. Kolmogorov-Smirnov tests were used to compare the observed and predicted age compositions of the production set.

The first two transects were drawn from the primordia to the edge of the ventral lobe. The third transect was drawn from the primordium to the distal edge of the otolith. The example is of an 8 year old 93 cm female blue grenadier captured on 29/6/98 from the west coast of Tasmania.

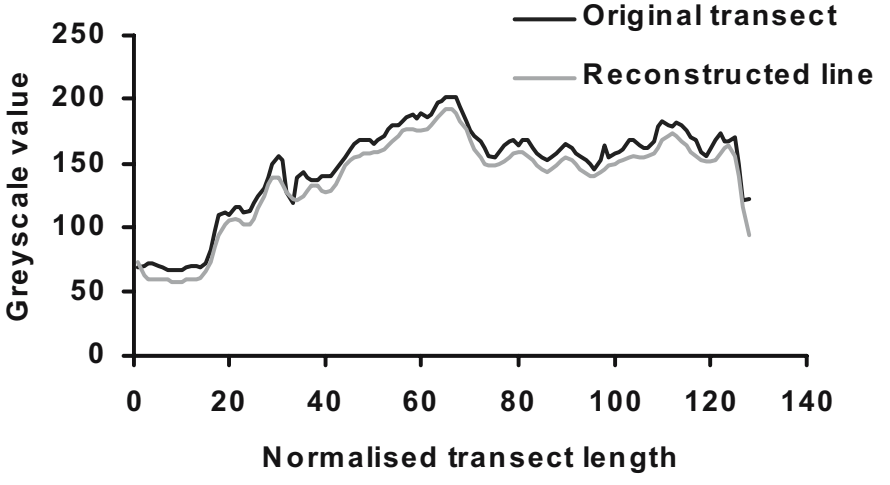


Figure 22.4. Original transect of grescale values and a transect reconstructed from an inverse discrete fast Fourier transform using twenty one complex numbers. The reconstructed line is displayed 10 greyscale values lower than the true value for ease of comparison.

22.5 Results

Trained networks achieved low APEs on the training sets. However, APEs were over 5% on the test sets, and close to 10% on the production sets for each of the three species (Table 22.2). Regression statistics showed significant bias in the estimated ages for *M. novaezelandiae* but not for the other two species. The fitted network models achieved R^2 values in excess of 0.8 for all species indicating that they explained a high level of the variation within the datasets.

For *A. butcheri* age bias plots show close agreement between mean predicted and observed ages although the standard errors are relatively wide for some age classes (Figure 22.5). The age composition plots show that the abundance of the strong year classes was underestimated (particularly the 3 and 9 year olds) and that of the adjacent weaker year classes was overestimated. Nevertheless, the Kolgomorov-Smirnov test found no significant difference between the distributions. Almost 50% of age estimates that were correct and almost 60% were within 1 year of the correct age (Table 22.2).

The age bias plots for *P. auratus* show good agreement between mean predicted and observed ages for the younger and older age classes but for the 7, 9 and 10 year old fish the neural network overestimated their age. The standard errors were wide for age classes with few individuals. The predicted and observed age compositions agreed closely for most ages up to 8 years. However, there was greater variability for the older age classes for which the neural network generally failed to match the age composition. However, the Kolgomorov-Smirnov test indicated that there was no significant difference between the distributions. The percentage of age estimates that were correct was similar to that for *A. butcheri*, but a much higher percentage was within one year of the correct age (Table 22.2).

The age bias plot for *M. novaezelandiae* also showed close agreement for the younger age classes, but poorer estimates for the older age classes. The mean age predicted by the neural networks did not differ greatly among age classes above 7 year old fish. This is reflected in the significant regression (the slope was significantly different from 1). However, there were relatively few fish in these older age classes in the production set so these differences did not produce great discrepancies between the observed and predicted age compositions, and the Kolmogorov-Smirnov test comparing these distributions was not significant. The sample for *M. novaezelandiae* produced the lowest percentage of correct age estimates but the highest percentage that were within 1 year of the correct age (Table 22.2).

Table 22.2. Model fits and bias and precision tests for each species.
APE=average percent error; Regression = */NS if either slope or intercept are/not significantly different from 1 or 0 respectively

Species	R^2	APE	Regression	% Correct	% Within 1 Year
A. butcheri	0.88	8.68	NS	49.5	78
P. auratus	0.81	9.18	NS	47.2	59.4
M. novaezelandiae	0.83	8.24	*	35.7	81.3

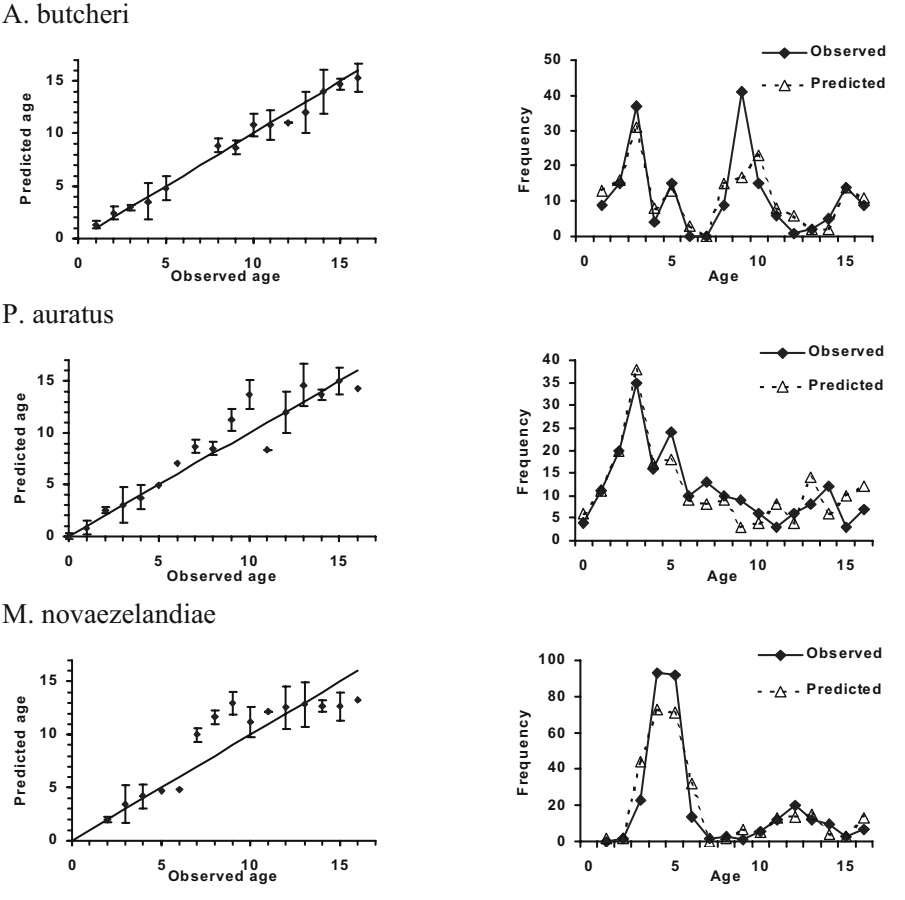


Figure 22.5. Age bias (left) and age composition (right) plots for each species. The age bias plots show mean (± 2 SE) predicted ages against observed age; the line shows equal age estimates.

22.6 Discussion

These results confirm the findings from the preliminary study of Robertson and Morison (1999) that neural networks offer a practical means to objectively estimate the age of fish. More confidence can be placed in the results presented here because of the larger sample sizes and the more rigorous testing procedures. The separation of training, test and production sets provides a test of the ability of

the trained networks to correctly classify new samples. These results will therefore provide a more accurate indication of the accuracy that could be expected if the method was implemented for ongoing production age estimation.

This study has also confirmed the value of three improvements suggested by Robertson and Morison (1999). Firstly, it has showed that the DFT can effectively capture the information from a transect across an otolith image. This robust and widely used transformation greatly reduces the number of data inputs and hence the complexity of the neural network model that needs to be constructed. The PNN models were fitted with less than half the number of inputs used in the pilot study, but were still able to include signal data from five transects rather than one, and incorporate auxiliary data as well. Lagardère and Troadec (1997) used a Fourier transformation for estimating the number of daily increments on the otoliths of larval sole *Solea solea*, but applied it to a demodulated signal. Further refinements of this approach are possible, so that the information in a greater amount of the image can be incorporated into future models.

Secondly, this study has shown that the biological data (fish length, otolith weight, and date of capture) can be incorporated into neural networks and contribute to their ability to estimate the age of samples. The relative importance of these auxiliary data sources to each model can be assessed from the magnitude of the individual scaling factors the trained network assigns to each input variable . For this study the most important variables varied for each species: for *A. butcheri* 9 of the 10 inputs with the highest scaling factors were Fourier transform harmonics; for *P. auratus* and *M. novaezelandiae* otolith weight, transect lengths, fish length and Fourier transform harmonics were all given high values. However, there was no clear point of demarcation of input variables with high or low scaling factors. The nature of the differences between the species were unexpected, especially since the otoliths of *A. butcheri* and *P. auratus* are similar in shape and clarity whereas those of *M. novaezelandiae* are more complex and difficult to interpret.

Thirdly, this implementation of neural networks has confirmed that PNNs are an appropriate network type for the problem of age estimation. The relative performance of PNNs compared with other types of neural networks was not explored here, and would be worthwhile, but the scaling factors they produce for each network input are one feature of PNNs that shows there utility for fish age estimation. They provide an ability to differentiate important inputs from those that contribute little discriminating power in a manner that is analogous to sensitivity tests. Sensitivity tests have been used to explore the relative importance of the data sources and the strength of their contribution to the response variable of interest (Walter *et al.* 2001). It does not add greatly to the complexity of the model to incorporate all the readily available data types into a model. Nevertheless, it would be useful to have a formal procedure for the identification of the minimum number of inputs needed to obtain satisfactory model predictions.

The PNNs showed an improvement over the results of the back-propagation models used in the pilot study (Robertson and Morison 1999) in producing acceptable results under more rigorous training and testing procedures. Results for *M. novaezelandiae* in particular were superior, as the earlier neural network

models failed to successfully estimate the age of this species. Despite the promising progress, the results are still below the performance of human readers who usually achieve APE values below 1% for *A. butcheri* and *P. auratus*, and below 5% for *M. novaezelandiae* (Morison *et al.* 1998a). Nevertheless, the PNNs correctly classified a much larger range of age classes than previous studies have managed (e.g. Welleman and Storbeck 1995; Troadec *et al.* 2000). Also, these results were achieved on a representative subset of otoliths and without the need for any *a priori* growth patterns or user intervention.

There are still several issues that would benefit from further investigation. Some of the apparent errors of the neural network age estimates may be due to mistakes made the human reader in estimating the age of the samples used to train the networks. Such errors may affect the ability of neural networks to train effectively or degrade their performance on the production set. This issue could only be effectively addressed by the use of all known age samples to train and test the network. However, such data sets are rare and difficult to obtain for many commercially exploited fish species.

One further development that is highly desirable is to test the performance of neural networks against more conventional multi-variate statistical methods. Few attempts have been made along these lines for age estimation in fish. (Boehlert 1985) used multiple linear regression methods but there are other well-established non-linear multi-variate statistical methods that could employed. Neural networks require such benchmarking to truly test the extent to which they represent a significant advancement in modelling for age estimation in fish or other ecological issues.

Acknowledgments

This work was supported by grants from the Australian Fisheries Research and Development Corporation (project numbers 96/136 and 98/105). Fisheries Victoria provided funds for the collection and ageing of *A. butcheri* and *P. auratus*. The Australian Fisheries Management Authority provided funds for the collection and ageing of samples of *M. novaezelandiae*. The assistance of many colleagues at the Marine and Freshwater Resources Institute is greatly appreciated, particularly Corey Green and Kyne Krusic-Golub for estimating ages, staff of the Bay and Inlet program for sample collection, and David Smith for comments on the draft manuscript. Steve Ward is thanked for his prompt and insightful suggestions for implementing the neural networks.

References

Beamish RJ, Fournier DA (1981) A method for comparing the precision of a set of age determinations. Journal of the Fisheries Research Board of Canada 36, 1395-1400

Beamish RJ, McFarlane GA (1983) The forgotten requirement for age validation in fisheries biology. Transactions of the American Fisheries Society(112), 735-43

Bennett JT, Boehlert G, Turekian KK (1982) Confirmation of longevity in Sebastes diploproa (Pisces: Scorpaenidae) from 210Pb/226Ra measurements in otoliths. Marine Biology 71, 209-215

Boehlert GW (1985) Using objective criteria and multiple regression models for age determination in fishes. Fisheries Bulletin U.S.A. 83, 103-117

Cailliet GM, Botsford LW, Brittnacher JG, Ford G, Matsubayashi M, King A, Watters DL, Kope RG (1996) Development of a computer-aided age determination system - evaluation based on otoliths of bank rockfish off California. Transactions of the American Fisheries Society 125(6), 874-888

Campana SE (2001) Accuracy, precision and quality control in age determination, including a review of the use and abuse of age validation methods (Review). Journal of Fish Biology 59(2), 197-242

Campana SE, Annand MC, McMillan JI (1995) Graphical and statistical methods for determining the consistency of age determinations. Transactions of the American Fisheries Society 124, 131-138

Campana SE, Thorrold SR (2001) Otoliths, increments, and elements: keys to a comprehensive understanding of fish populations? Canadian Journal of Fisheries & Aquatic Sciences 58(1), 30-38

Chang WYB (1982) A statistical method for evaluating the reproducibility of age determination. Canadian Journal of Fisheries and Aquatic Sciences 39, 1208-1210

Fletcher WJ (1995) Application of the Otolith Weight - Age Relationship For the Pilchard, Sardinops Sagax Neopilchardus. Canadian Journal of Fisheries & Aquatic Sciences 52(4), 657-664

Francis RICC, Paul LJ, Mulligan KP (1992) Ageing of adult snapper (Pagrus auratus) from otolith annual ring counts: validation by tagging and oxytetracycline injection. Australian Journal of Marine and Freshwater Research 43, 1069-1089

Gröger J (1999) A theoretical note on the interpersonal correction of age readings by means of calibration techniques. Archive of Fishery Marine Research 47(1), 77-101

Kalish JM (1993) Pre- and post-bomb radiocarbon in fish otoliths. Earth and Planetary Science Letters 114, 549-554

Kalish JM, Johnston JM, Smith DC, Morison AK, Robertson SG (1997) Use of the bomb radiocarbon chronometer for age validation in the blue grenadier Macruronus novaezelandiae. Marine Biology 128(4), 557-563

Kimura DK, Lyons JJ (1991) Between reader bias and variability in the age-determination process. Fishery Bulletin 89, 53-60

Lagardere FTH (1997) Age estimation in common sole Solea solea larvae - validation of daily increments and evaluation of a pattern recognition technique. Marine Ecology-Progress Series 155, 223-237

Macy WKI (1995) The application of digital image processing to the aging of ling-finned squid, Loligo pealei, using the statolith. In 'Recent Developments In Fish Otolith Research'. (Eds Secor, D. H., Dean, J. M., and Campana, S. E.) pp 283-302. (University of South Carolina Press: Colombia.)

Masters T (1993) Practical Neural Network Recipes in C++. (Academic Press Inc.: San Diego.)

Masters T (1995) Advanced algorithms for neural networks. A C++ sourcebook. 437 pp. (John Wiley and Sons: New York.)

Morison AK, Coutin PC, Robertson SG (1998a) Age determination of black bream, Acanthopagrus butcheri (Sparidae), from the Gippsland Lakes of south-eastern Australia indicates slow growth and episodic recruitment. Marine and Freshwater Research 49, 491-98

Morison AK, Robertson SG (1997) Automatic ageing of fish from otoliths: a pilot study. Final report to FRDC for Project #96/136 (Marine and Freshwater Resources Institute: Queenscliff.)

Morison AK, Robertson SG, Smith DC (1998b) An integrated system for production fish aging: image analysis and quality assurance. North American Journal of Fisheries Management 18, 587-98

Punt AE, Smith DC, Thomson RB, Haddon M, He X, Lyle JM (2001). Stock assessment of the blue grenadier Macruronus novaezelandiae resource of south-eastern Australia. Marine and Freshwater Research 52, 701-717

Richards LJ, Schnute JT, Kronlund AR, Beamish RJ (1992) Statistical models for the analysis of ageing error. Canadian Journal of Fisheries and Aquatic Sciences 49, 1801-1815

Robertson SG, Morison AK (1999) A trial of artificial neural networks for automatically estimating the age of fish. Marine and Freshwater Research 50, 73-82

Takashima Y, Takada T, Matsuishi T, Kanno Y (2000) Validation of auto-couning method by NIH Image using otoliths of white-spotted char Salvelinus leucomanenis. Fisheries Science 66, 515-520

Troadec H (1991) Frequency demodulation on otolith numerical images for the automation of fish age estimation. Aquatic Living Resources 4, 207-219

Troadec H, Benzinou A, Rodin V, Le Bihan J (2000) Use of deformable template for two-dimensional growth ring detection of otoliths by digital image processing: Application to plaice (Pleuronectes platessa) otoliths. Fisheries Research 46, 155-163

Walter M, Recknagel F, Carpenter C, Bormans M (2001) Predicting eutrophication effects in the Burrinjuck Reservoir (Australia) by means of the deterministic model SALMO and the recurrent neural network model ANNA. Ecological Modelling 146, 1-3, 97-113

Welleman HC, Storbeck F (1995) Automatic ageing of plaice (Pleuronectes platessa L.) otoliths by means of image analysis. In 'Recent Developments in Fish Otolith Research'. (Eds Secor, D. H., Dean, J. M., and Campana, S. E.) pp 271-282. (University of South Carolina Press: Columbia.)

Williams T, Bedford BC (1974) The use of otoliths for age determination. In 'Ageing of Fish'. (Eds Bagenal, T. B.) pp 114-123. (Unwin Brothers: Old Woking.)

Worthington DG (1995) Variation in the relationship between otolith weight and age - implications for the estimation of age of two tropical damselfish (Pomacentrus moluccensis and P. wardi). Canadian Journal of Fisheries & Aquatic Sciences 52(2), 233-242

Chapter 23

Pattern Recognition and Classification of Remotely Sensed Images by Artificial Neural Networks

G.M. Foody

23.1 Introduction

Pattern recognition is concerned with a range of information processing issues associated with the description or classification of measurements. It is based on a broad and often loosely related body of literature and techniques (Schalkoff, 1992). Although statistical and structural (syntactic) approaches have dominated the subject there has been a growing interest in the use of neural networks for pattern recognition applications (Schalkoff, 1992; Bishop, 1995). This is particularly evident in relation to pattern recognition applications in remote sensing. Remote sensing is often used to derive information on the environment (Campbell, 2002; Lillesand *et al*., 2004). For example, satellite remote sensors are commonly used to provide images of the Earth's surface that may be analysed to yield information on a diverse range of issues of ecological significance. This includes information on a variety of important environmental phenomena including land cover and its dynamics (e.g. deforestation), vegetation productivity and yield, soil water content, water quality and variation in surface temperature. Remote sensing has, therefore, the potential to provide information to support ecological research, particularly that addressing macro or coarse spatial scale issues (Roughgarden *et al*., 1991; Kasischke *et al*., 1997; Lucas and Curran, 1999; Hall, 2000). The value of remote sensing as a source of information on the environment is, however, sometimes limited by the image analysis techniques used. Traditionally, statistical techniques have been used widely in the analysis of remotely sensed data. However, in common with other fields of study, including ecology, the last 10-15 years has witnessed a rapid growth in the use of neural networks (Atkinson and Tatnall, 1997; Lek *et al*., 2000). This chapter aims to report on some of the main application of neural networks in remote sensing and indicate topics where further development may be expected to occur.

23.2 Neural Networks in Remote Sensing

Of the many pattern recognition applications, neural networks have been most widely used in remote sensing for image classification and regression-type analyses. They have been adopted increasingly as a result of their freedom from restrictive assumptions as well as practical demonstrations of their ability to commonly provide more accurate outputs than conventional methods.

23.2.1 Classification Applications

Remotely sensed data are extremely attractive for thematic mapping applications. This is mainly because the data have a map-like format, provide complete and continuous coverage of large areas inexpensively and are relatively consistent. Consequently, remote sensing has become a major source of thematic maps such as those depicting land cover.

Thematic mapping is typically based on some sort of classification analysis. Two broad types of classification are used, supervised and unsupervised (Schowengerdt, 1997; Mather, 1999). The unsupervised classifiers are essentially clustering techniques. These are generally used to search for natural spectral classes in the remotely sensed data. This type of approach can be useful in exploratory analyses and where a basic generalisation of the data set is required. Deriving meaningful labels for the derived clusters is, however, sometimes difficult. More commonly, therefore, supervised classification techniques are used. With this type of classification the classes of interested are known at the outset. The classification algorithm is trained to recognise the remotely sensed response of each class and then applied to the image in order to allocate each pixel in the image to the class with which it has the greatest similarity (Campbell, 2002; Mather, 2004). This type of approach has been widely used in studies of land cover and land cover change. There are, however, many problems that often limit the accuracy of classification analyses. Thus, the considerable potential of remote sensing as a source of land cover information is often not fully realised (Townshend, 1992; Wilkinson, 1996; Estes *et al*., 1999). Many factors may be responsible for this situation, ranging from the characteristics of the remote sensor (e.g. its spectral, radiometric, temporal and spatial resolutions), the nature of the classes (e.g. level of class detail, degree of class heterogeneity) and the classification methods used (Foody and Arora, 1997; Scepan, 1999; Estes *et al*., 1999; Loveland *et al*., 1999; Friedl *et al*., 2000). The latter issue has been the focus of considerable attention. Recently attention has focused on neural network based approaches for image classification which offer considerable advantages over traditional methods (e.g. statistical approaches such as the maximum likelihood classification), particularly with regard to the freedom from restrictive assumptions about the data sets (Fischer *et al.*, 1997; Fischer, 1998). Comparative

analyses have also generally shown that neural networks may be used to classify remotely sensed data more accurately than conventional statistical approaches, particularly if the data are incompatible with the assumed model that underlies statistical classifications (Benediktsson *et al*., 1990; Peddle *et al*., 1994; Paola and Schowengerdt, 1995).

23.2.2 Regression Applications

Commonly remotely sensed data have been used to derive estimates of environmental variables. Thus, for example, remotely sensed data have been used to estimate a range of environmental phenomena including the biomass of forests, moisture content of soils and the temperature and sediment content of water bodies. This is often the only feasible way of deriving information on the variables over large areas and the only practical source of spatial data to drive some ecosystem simulation models (Lucas and Curran, 1999). A variety of methods may be used to derive the estimates of environmental variables but regression techniques have proved popular (e.g. Lawrence and Ripple, 1998 Curran *et al*., 2001). As with the classification of remotely sensed data, a fundamental problem is that the assumptions that underlie regression analysis may not be satisfied, leading to significant error and misinterpretation. Neural networks, however, have considerable potential as non-parametric alternative to regression analysis and in recent years have been used increasingly in the extraction of environmental information from remotely sensed data (e.g. Baret, 1995; Jin and Liu, 1997; Foody *et al*., 2001). Similarly, neural networks have been widely used in the related application of model inversion (e.g. Wang and Dong, 1997).

23.3 The Neural Networks Used in Remote Sensing

A variety of neural networks have been used in remote sensing. This section aims to briefly outline the key features of the most widely used networks. Emphasis will be placed upon the feedforward networks, such as the multi-layer perceptron (MLP), radial basis function (RBF) probabilistic neural network (PNN) and generalised regression neural networks (GRNN), that have been used most widely in remote sensing. Other network types that have been used less frequently but for which there is considerable scope for further application in remote sensing are also discussed briefly.

23.3.1 Feedforward Neural Networks

Feedfoward networks comprise a set of simple processing units arranged in layers that are interconnected and can, once trained, be used for classification and regression applications (Figure 23.1). The number of input and output units is determined by the characteristics of the remotely sensed data used (e.g. the spectral wavebands) and the desired output (e.g. number of classes in a classification). In most studies, there is, for example, one input unit associated with each spectral waveband in the remotely sensed imagery to be used and one output unit associated with each class to be classified or variable to be estimated, although other architectures are possible (e.g. if using thermometer or spread encoding of data). The nature of the hidden layers and of the interconnections between units in different layers, however, differs substantially between the different types of feedforward network.

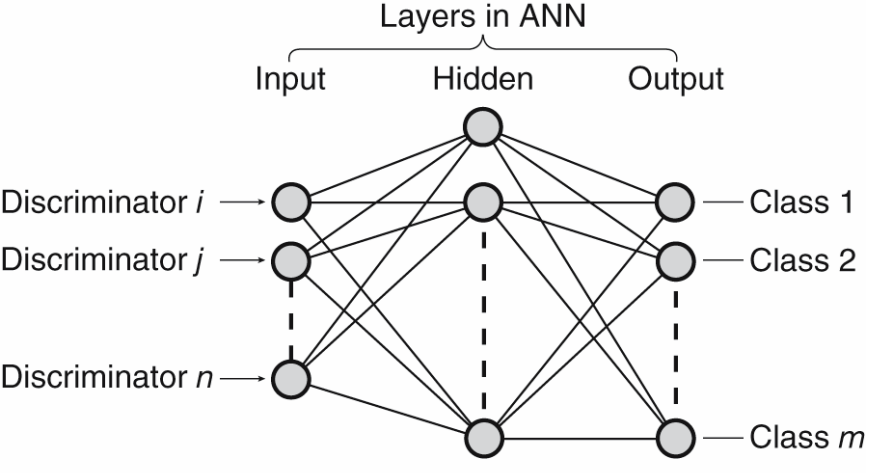


Figure 23.1. A basic feedforward neural network with a single hidden layer. The lines connecting the units between layers represent the weighted connections. The network illustrated uses *n* input variables (e.g. the 6 non-thermal channels of the Landsat TM) to derive an *m* class classification. For regression type applications there is normally only a single output (e.g. a prediction of an environmental variable such as forest biomass).

23.3.1.1 Multi-Layer Perceptron (MLP)

The MLP is probably the most widely used type of neural network in remote sensing (Day, 1997; Atkinson and Tatnall, 1997). With an MLP, the number of hidden units and hidden layers is typically determined after a series of trial runs, although some heuristics may be used to help guide the selection of the network architecture (e.g. Wang, 1994; Kanellopoulos and Wilkinson, 1997). In general, the more complex the problem the larger the network, in terms of hidden units and layers, required for its solution. Each unit in a layer of the MLP network is connected to every unit in the adjacent layer(s) of the network by a weighted connection (Figure 23.1). The units undertake very simple analyses. For example, a unit in the hidden layer simply formulates the weighted sum of all of its inputs and passes this through its transformation function to generate the output that is then propagated to units in the next layer. The input to a unit, such as one of the hidden units in Figure 23.1, is calculated from,

$$net_h = \sum_{a=1}^{N} w_{ha} o_a \tag{23.1}$$

where o_a is the magnitude of the output from unit a in the previous layer that contains N units and w_{ha} the weight of the interconnection channel between units a and h. This net input (net_h) is then transformed by the unit's activation function to produce an output for the unit (Schalkoff, 1992). Typically a sigmoidal activation function,

$$o_h = 1/(1+\exp(-\lambda net_h)) \tag{23.2}$$

where λ is a gain parameter, that is often set to 1.0, is used. Although each individual unit in the network performs relatively simple analyses, the network as a whole may be used to solve complex problems. This is achieved by setting the weights connecting the units to values that enable the network to accurately predict class membership or estimate the value of the phenomenon of interest from the remotely sensed data presented to it. The magnitude of each weighted connection is determined via an iterative training procedure using a learning algorithm such as backpropagation or quickpropagation (Davalo and Naim, 1991; Schalkoff, 1992; Bishop, 1995). Training begins with the magnitude of the weighted connections set at randomly determined values. The training sample is then passed through the network and, as the desired output of the network is known for each case in the training sample, the error in the network's predictions may be calculated. This derived error is then effectively passed backwards through the network with the magnitude of the weights connecting the units adjusted in relation to the error magnitude. Typically the weight adjustment is made with an application of a function such as,

$$\Delta w_{ha}(f) = -\eta \delta_h o_a + \alpha \Delta w_{ha}(f\text{-}1) \quad (23.3)$$

where f is the iteration number, δ_h a computed error and η and α are parameters which define the learning rate and momentum which may facilitate network learning (Schalkoff, 1992). The process of entering the training data, calculating the network's error and adjusting the magnitude of the weighted connections between units in relation to the error magnitude continues until the network is capable of identifying class membership or predicting the magnitude of the phenomenon of interest accurately. The selection of an appropriate point at which to stop the training of the network is a difficult but important task. The analyst must seek to avoid the problems of under- and over-fitting the network to the training data and frequently achieves this through the use of a verification set. The sample of cases contained in the verification set are used to evaluate the generalisation ability of the network but are not used directly in the training of the network. By classifying the verification set at intervals during the training of the network it is usually possible to identify a suitable point at which to stop training. In this way dependence on the potentially mis-leading training or learning error, calculated on the training set, is reduced and a guide to the generalisation ability of the network is provided.

23.3.1.2 Radial Basis Function (RBF)

RBF networks have been used less frequently in remote sensing than the MLP. The RBF network, however, has considerable potential in remote sensing studies (Fischer *et al*., 1997; Rollet *et al*., 1998; Bruzzone and Prieto, 1999; Foody, 2004a). While the MLP can have one or more hidden layers the RBF has only a single hidden layer of units. The units contained in this hidden layer are very different to those used in the MLP. The hidden units in the RBF network use a radial basis transformation function which responds to only a small region of the input space upon which it has been centred (Bishop, 1995). Each of the radial basis functions has two key parameters that describe the location of the function's centre and its width. The hidden unit measures the distance between an input data vector and the centre of its radial basis function. This, together with the function's width is used to derive the input to the hidden unit from which the unit's output may then be calculated. This radial basis function has its peak at zero distance and declines with increasing distance from the centre. Consequently, the output of the radial unit is 1.0 if the input data vector lies on the function's centre but declines the greater the distance between the input vector and function's centre until it is 0.0. The activation level of a basis function is, therefore, constant on concentric circles around its centre and hence the feature space is partitioned into hypersheres (Bishop, 1995). This is very different to the basic MLP with a single hidden layer that partitions the entire feature space with hyperplanes along which the activation level of the hidden units is constant. Since the RBF units respond to local regions it is common for a RBF network to require a more complex

architecture, comprising more hidden units, than a MLP network constructed for the same problem.

The RBF model provides a smooth interpolating function in which the number of basis functions and so hidden units required is a function of the complexity of the problem in-hand rather than the size of the data set (Bishop, 1995). As there is only a single hidden layer there are only two sets of weights in the network, one connecting the hidden layer to the input layer and the other connecting the hidden layer to the output layer. Those weights connecting to the input layer contain the parameters of the basis functions. The weights connecting the hidden layer to the output layer are used to form linear combinations of the activations of the basis functions (hidden units) to generate the network outputs. Since the hidden units are non-linear, the outputs of the hidden layer may be combined linearly and so processing is rapid. The output of the network is derived from,

$$y_k(x) = \sum_{j=1}^{M} w_{kj}\ \phi_j(x) + w_{ko} \tag{23.4}$$

where M is the number of basis functions, x the input data vector, w_{kj} represents a weighted connection between the basis function and output layer and ϕ_j is the non-linear function of unit j, which is typically a Gaussian of the form

$$\phi_j(x) = \exp(-|x-\mu_j|^2 / 2\sigma_j^2) \tag{23.5}$$

in which μ_j and σ_j are the parameters specifying the basis function's centre and width respectively.

Training a RBF network involves two distinct stages. In the first stage, the basis functions are determined using an unsupervised analysis. This effectively defines the weights connecting the units in the input and hidden layers. The determination of the basis function parameters in this first stage of training the RBF network may also make use of unlabelled cases (Bishop, 1995). This can be advantageous in remote sensing applications as ground data are often scarce or costly to obtain but the remotely sensed data set is generally voluminous (Shahshanhani and Landgrebe, 1994; Fardanesh and Ersoy, 1998). In the second stage of training, the weights connecting the hidden and output layers are derived using a linear supervised method (Bishop, 1995).

23.3.1.3 Probabilistic Neural Networks (PNN)

The PNN is used only for classification problems and is typically very much larger than an MLP or RBF network designed to undertake the same task. With the PNN network, estimates of the probability density functions (p.d.f.) of the possible classes are derived and used to select the most probable class of

membership for each case (Specht, 1990). Rather than assume a certain form for the p.d.f. (e.g. a normal distribution) a kernel-based approximation is commonly used (Bishop, 1995). With this, simple functions are located at each available case and added together to derive an estimate of the overall p.d.f. This kernel-based approach to p.d.f approximation has similarities to aspects of the RBF networks. Indeed, radial units are contained in the hidden layer of a PNN. In total, the PNN contains at least three layers with the input and output layers performing similar functions to the other network types. The radial units of the PNN essentially copy the training data and hence the number of hidden radial units is equal to the size of the training set. PNN networks are consequently typically very much larger than MLP and RBF networks designed for the same problem. Each radial unit models a Gaussian function centred on the relevant training case and is connected to the unit in the output layer associated with the actual class of membership for that training case. Thus each output unit is connected to only the radial units associated with the same class with a zero connection to all other radial units. In the output units, the sum of all the responses of the radial units belonging to the relevant class is derived and these are proportional to the kernel based estimates of the p.d.f.'s of the defined classes. These outputs may be normalised to sum to unity to provide a probabilistic output and each case of previously unknown membership may be allocated to the class with which it has the highest probability of membership. This type of network can be rapid to train but, because of its size, slow to apply to large data sets. It has a major advantage over the other network types of being able to readily accommodate prior information into the analysis, which is often advantageous in remote sensing applications (Strahler, 1980; Mather, 2004) and can be used to increase classification accuracy (Foody, 2001).

23.3.1.4 Generalised Regression Neural Network (GRNN)

The GRNN is a regression (Bayesian) network that has some similarities with the PNN. The GRNN, however, has four layers and is used only for regression-type problems. First, is the input layer as normal. Second, a layer of radial units that are used to provide a representation of the centres of clusters identified from the training sample. The radial layer is typically large, but usually smaller than the size the training sample, and is trained using a clustering algorithm. The third layer in the network is a layer of regression units. Two types of unit are encountered in this layer, one calculates the desired regression outputs while the other calculates the probability density. In total there is always one more unit in this third layer than in the output layer. The fourth layer is the output layer that integrates the outputs from the regression layer to provide the networks predictions. Relative to MLP and RBF networks constructed for the same problem, a GRNN typically has a much larger architecture (e.g. Foody *et al*., 2001). Because of the large size of this type of network it is typically slow to apply to large data sets, although it does train rapidly.

23.3.1.5 Other Network Types

A range of other networks has been used in remote sensing (e.g. Corne *et al*., 2004). Three in particular are worthy of brief discussion. These are the Hopfield, ART and Kohonen neural networks. Each is very dissimilar to the feedforward networks that have been described above.

The Hopfield neural network belongs to a set of recurrent networks. Unlike the feedforward networks, in which information is essentially propagated in a single direction, the outputs of units in a Hopfield network are fed back into the inputs. The Hopfield network is, therefore, a fully connected network (Aleksander and Morton, 1990; Davalo and Naim, 1991). It was used initially as a contents addressable (associative) memory device but is also able to solve complex combinatorial problems (Cote and Tatnall, 1997). The Hopfield neural network has been used in a range of applications in remote sensing including the tracking of clouds (Cote and Tatnall, 1997) and in the refinement of soft/fuzzy image classifications in order to effectively sharpen the resolution of thematic maps (Tatem *et al*., 2001; 2003).

Adaptive resonance theory (ART) provides the foundation for real-time networks that solve the stability-plasticity problem that handicaps the use of other approaches such as the MLP (Carpenter *et al*., 1999a; Gopal *et al*., 1999; Gamba and Houshmand, 2001). That is, the ART networks are able to perform rapid, stable on-line learning, recognition and prediction. ART networks may be adapted to accommodate fuzzy logic and have been used in supervised image classification applications (Gopal *et al*., 1999; Carpenter *et al*., 1999b).

The Kohonen network belongs to a set of self-organising competitive learning systems. Sometimes referred to in the literature as a self-organising map (SOM) or self-organising feature map (SOFM), the Kohonen network has been used commonly for unsupervised classification. The network uses unsupervised learning to provide a topologically ordered output that displays the relative similarity of cases entered to it. The network consists of just two layers of units, input and output. The output is a low, typically 2, dimensional array of units. Each unit in the output layer is fully connected to all adjacent units in that layer as well as to all of the input units. The lateral connection of units in the output layer aids competitive learning such that similar cases cluster together in the output array and are associated with a different region of the output array than dissimilar cases. This network has been used for unsupervised classification (Poth *et al*., 2001), although if training data are available it may also be used to label cases (e.g. Ito and Omato, 1997; Ji, 2000).

23.4 Current Status

The potential of neural networks for the provision of accurate environmental information from remotely sensed data has been demonstrated in numerous studies. With regard to both classification and regression type problems this has been largely demonstrated with the MLP network.

Many studies have demonstrated that neural networks can generally be used to classify remotely sensed data at least as, but usually more, accurately than conventional statistical classifiers (e.g. Benediktsson *et al*., 1990). For example, Peddle *et al*. (1994) provide a comparative evaluation of neural network, maximum likelihood and evidential reasoning classifications of alpine land cover. The results show that for a variety of classification scenarios based upon SPOT HRV and ancillary topographic data that, in general, the highest accuracy was obtainable when a neural network was used to classify the data. For example, in one set of analyses it was apparent that the accuracy of a maximum likelihood classification declined with the addition of ancillary information on topography from 75% to 32% approximately while the corresponding accuracies derived from a neural network showed and increase from 75% to 90% approximately. This result is significant in that it shows the neural network was, unlike the conventional statistical classification, able to constructively use the ancillary information and derive a significantly higher classification accuracy.

The MLP has also been used successfully in regression applications. Many studies have focussed on the estimation of biophysical properties from remotely sensed data (e.g. Weiss and Baret, 1999; Kimes *et al*., 2000; Boyd *et al*., 2002). Studies that illustrate the value of neural networks for this type of application include those focused on the estimation of leaf area index (Smith, 1993) and forest biomass (Foody *et al*., 2001). In terms of relative performance against standard regression analysis, neural networks have been shown to be able to derive more accurate predictions. For example, in the estimation of tropical forest biomass from Landsat TM data, Foody *et al*. (2001) derived correlation coefficients between the biomass predicted from a multiple regression analysis and a MLP neural network with that derived from field measurement of r=0.50 and r=0.80 respectively.

Although the MLP is the most widely used neural network in remote sensing data analysis attention is increasingly turning to alternative network types (e.g. Corne *et al*., 2004). For supervised classification applications, perhaps the most common use of neural networks in remote sensing, attention has recently turned increasingly to the RBF (Fischer *et al*., 1997; Rollet *et al*., 1998; Bruzzone and Prieto, 1999), PNN (Foody, 2001) and ART type networks (Carpenter *et al*., 1999a; Gopal *et al*., 1999; Liu *et al*., 2004) networks which may sometimes be more appropriate than the MLP.

23.4.1
An Example of Neural Networks for Classification

To help appreciate the potential of artificial neural networks for use in remote sensing a series of classifications using the MLP, RBF and PNN outlined above were undertaken. These classification analyses used multi-spectral imagery acquired by an airborne thematic mapper sensor of an agricultural site in the UK; details on the data set and test site may be found in Foody and Arora (1997). Briefly, the data set comprised the remotely sensed response in 11 spectral wavebands for 600 pixels sampled from the imagery. These 600 pixels were extracted through a stratified random sampling design that ensured that 100 pixels were extracted for each of the six main crop classes found at the test site.

The data set was divided into equally sized but independent training and testing sets. The training set was, however, further divided into a classical training set and a verification set to help guide the parameterization of the neural networks. For the purpose of this example, 10 pixels of each class were used to form the verification sample. A software package that aims to optimally design neural networks was then used to generate appropriate MLP, RBF and PNN classifiers, using the accuracy of the verification set as a guide to network selection. In each case the starting point was that n=11 and m=6 (Figure 23.1), although it was possible for relatively uninformative inputs, and so units, to be ignored.

In addition to the neural networks, a standard quadratic discriminant analysis was used to derive a conventional maximum likelihood classification. This was a stepwise dicriminant analysis, using Wilks' lambda as the variable selection criterion, in which the entire training sample was used as the training set (there was no need for a verification sample).

Each of the 4 classifications, therefore, used the same training and testing samples, although the division of the training sample into training and verification sets does slightly reduce the direct comparability of the analyses. A summary of the networks generated and of the classifications derived is given in Table 23.1.

Table 23.1. Summary of the classifiers and the accuracy with which they classified the independent testing set (DA = discriminant analysis). The neural network architecture is described in terms of the number of input:hidden:output units and this, along with other network parameters, was determined by an empirical procedure. Note that discriminating variables (wavebands) that contributed little could be ignored in each analysis (the data some wavebands were excluded in neural network and the discriminant analysis)

Classifier	Architecture	Accuracy (%)
MLP	11:26:6	94.33
RBF	9:27:6	90.00
PNN	7:240:	90.66
DA	Not applicable	91.33

Table 23.2. Confusion matrices from the classifications: (a) MLP, (b) RBF, (c) PNN and (d) DA. Note in each matrix the actual class of membership is shown in the columns and the predicted class of membership in the rows. Classes are sugar beet (SB), wheat (W), barley (B), carrots (C), potatoes (P) and grass (G).

Class	SB	W	B	C	P	G	Total
SB	45	1	2	0	2	0	50
W	2	48	2	0	0	2	54
B	0	1	46	0	0	0	47
C	1	0	0	48	0	0	49
P	2	0	0	2	48	0	52
G	0	0	0	0	0	48	48
Total	50	50	50	50	50	50	300

(a) MLP

Class	SB	W	B	C	P	G	Total
SB	42	3	1	0	0	0	46
W	4	43	2	0	2	1	52
B	1	4	45	0	0	0	50
C	1	0	0	47	0	4	52
P	2	0	0	3	48	0	53
G	0	0	2	0	0	45	47
Total	50	50	50	50	50	50	300

(b) RBF

Class	SB	W	B	C	P	G	Total
SB	41	5	0	0	0	0	46
W	4	42	2	0	2	0	50
B	2	2	47	0	0	0	51
C	0	1	0	47	0	3	51
P	2	0	0	3	48	0	53
G	1	0	1	0	0	47	49
Total	50	50	50	50	50	50	300

(c) PNN

Class	SB	W	B	C	P	G	Total
SB	45	0	0	0	0	0	45
W	1	46	0	0	0	0	47
B	0	4	50	0	0	0	54
C	1	0	0	45	1	1	48
P	3	0	0	5	48	9	65
G	0	0	0	0	1	40	41
Total	50	50	50	50	50	50	300

(d) DA

It was apparent that the six crop classes exhibited a high degree of separability in the remotely sensed data set, with the accuracy with which the testing set was classified =90% for each method. The highest accuracy, 94.33%, was observed for the classification by the MLP. Moreover, the difference in accuracy between the MLP classification and the least accurate classification, derived from the RBF, was statistically significant (p<0.05, difference between proportions test; Foody, 2004b).

A more detailed assessment of the 4 classifications can be made from their associated confusion matrices (Table 23.2). It is evident from these matrices that while the differences in overall classification accuracy between the 4 classifications were relatively small there were substantial differences in the patterns of class allocations. The quality of a particular classification may, therefore, vary with the specific requirements of a user. In, general, however, the focus is on overall classification accuracy and consequently, in this example, the MLP would be viewed as the most accurate and useful classifier. It is, it also worth noting that all of the neural network classifiers yielded classifications in which the number of cases allocated to each class was relatively constant and, linked in part to this, the individual class accuracies were relatively similar. This is often a desirable feature in classifications and is not the case with the discriminant analysis. Note for example, that the range of the number of cases allocated to the classes was 24 (65-41) for the discriminant analysis but only 7 for each of the neural networks. The accuracy with which each individual class was classified also varied between the techniques but was most variable for the discriminant analysis. Unless interest was on a specific class, this would tend to make the neural network classifications more attractive to many users. It is possible, however, for a specific class to be classified most accurately by a classifier with low overall accuracy or variable individual class accuracies so the evaluation of the classifications is very dependent on the application in-hand. Overall, however, the results highlight the value of neural networks for accurate classification of remotely sensed data.

23.4.2 Concerns with Neural Networks

Despite the demonstrated potential of neural networks for classification and regression applications a variety of concerns and problems have been noted. For example, the difference in accuracy relative to that obtainable from a conventional approach is not always large (as in the example above), it can be difficult to understand the results derived and the analysis is based upon a set of subjective decisions (Wilkinson, 1997; Foody, 1999a). Issues such as the parameterization of a network as well as the size and nature of the training set have a marked impact on classification accuracy and other approaches may be more appropriate (Zhuang *et al*., 2004; Foody *et al*., 1995; Staufer and Fischer, 1997; Foody, 1999b;

Kavzoglu and Mather, 2003; Foody and Mathur, 2004; Pal and Mather, 2004). Moreover, if the assumptions underlying a model based technique are satisfied it has been suggested that that technique be used instead of the neural network in order to exploit fully advantage conveyed by the underlying model. Sometimes the problems encountered with neural networks relate to general issues. For example, although a neural network classification may make no assumptions about the data sets used the classification analysis is founded on a set of assumptions. Key amongst these are the commonly made assumptions that the classes have been defined exhaustively and are discrete and mutually exclusive. While these assumptions can be satisfied in demonstration projects that have shown the potential of neural networks they are less satisfiable in 'real world' studies. Thus the accuracy of neural network analyses may still be insufficient for some users (Wilkinson, 1996). Frequently, the problem here is that neural networks have generally been used to derive conventional 'hard' classifications (in which each pixel is allocated firmly to a single class). This is often inappropriate due to the presence of mixed pixels, that typically dominate remotely sensed data sets used in macro scale studies (Foody *et al*., 1997; Campbell, 2002). Moreover, in many studies the classes are continuous rather than discrete. Additionally, the set of classes may not have been defined exhaustively. There are, however, means of reducing these problems. For example, the problems associated with continuous classes and mixed pixels may be reduced by using the network to derive a soft or fuzzy classification rather than the conventional hard classification (Foody, 1996). Fuzzy classifications have been derived using a range of networks, notably the MLP (Foody, 1999c) and ARTMAP (Gopal *et al*., 1999). These have been found to provide accurate estimates of sub-pixel class composition and provide a richer thematic representation that the conventional hard classification. Additionally, using a Hopfield neural network the sub-pixel fractions derived from a soft classification may be located within the area represented by a pixel to derive an enhanced or super resolution thematic representation (Tatem *et al*., 2001; 2003). The effect of a non-exhaustively defined set of classes may also be reduced by post-classification thresholding of the network outputs or use of a network such as the RBF that partitions feature space locally and so is less prone to untenable extrapolation than the widely used MLP (Vasconcelos *et al*., 1995; Foody, 2001).

23.5 Conclusions

Neural networks are powerful general purpose computing tools. They have become popular in the analysis of remotely sensed data, particularly for classification and regression-type problems in which they have often been demonstrated to extract information more accurately than conventional methods. Although not free from problems, it seems likely that neural networks will be used

increasingly in ecological research using remote sensing. Moreover, as some of the problems encountered in use of neural networks arise from a tendency to focus upon the MLP only it is likely that there will be a greater use of other network types. In addition, it is expected that the range of applications of neural networks in remote sensing will broaden. Applications in which neural networks have already been used and increased usage may be expected include: image pre-processing (e.g. geometric, atmospheric and radiometric correction), stereo-matching imagery, image compression, feature extraction, map generalisation, multi-source data analysis, data fusion and image sharpening (e.g. Day, 1997; Foody, 1999a). Thus while neural networks have rapidly become established in remote sensing it is likely that they will be used increasingly and in a broader range of activities that will help exploit more fully the potential of remote sensing as a useful tool in ecological research.

Acknowledgements

This article draws upon material presented in a series of earlier papers and so has benefited from comment and discussion related to earlier work that is greatly appreciated.

References

Aleksander I, Morton H (1990) An Introduction to Neural Computing, Chapman and Hall, London.

Atkinson PM, Tatnall ARL (1997) Neural networks in remote sensing, International Journal of Remote Sensing, 18, 711-725.

Baret F (1995) Use of spectral reflectance variation to retrieve canopy biophysical characteristics, In: Danson FM, Plummer SP (eds) Advances in Environmental Remote Sensing, Wiley, Chichester, pp. 33-51.

Benediktsson JA, Swain PH, Ersoy OK (1990) Neural network approaches versus statistical methods in classification of multisource remote sensing data, IEEE Transactions on Geoscience and Remote Sensing, 28, 540-551.

Bishop CM (1995) Neural Networks for Pattern Recognition Oxford University Press, Oxford.

Boyd DS, Foody GM, Ripple WJ (2002) Evaluation of approaches for forest cover estimation in the Pacific Nortwest, USA using remote sensing, Applied Geography, 22, 375-392.

Bruzzone L, Prieto DF (1999) A technique for the selection of kernel-function parameters in RBF neural networks for classification of remote sensing images, IEEE Transactions on Geoscience and Remote Sensing, 37: 1179-1184.

Campbell JB (2002) Introduction to Remote Sensing, third edition, Taylor and Francis, London.

Carpenter GA, Gopal S, Macomber S, Martens S, Woodcock CE (1999a) A neural network method for mixture estimation for vegetation mapping, Remote Sensing of Environment, 70, 138-152.

Carpenter GA, Gopal S, Macomber S, Martens S, Woodcock CE, Franklin J (1999b) A neural network method for efficient vegetation mapping, Remote Sensing of Environment, 70, 326-338.

Corne SA, Carver SJ, Kunin WE, Lennon JJ, van Hees WWS (2004) Predicting forest attributes in southeast Alaska using artificial neural networks, Forest Science, 50, 259-276.

Cote S, Tatnall ARL (1997) The Hopfield neural network as a tool for feature tracking and recognition from satellite sensor images, International Journal of Remote Sensing, 18, 871-885.

Curran PJ, Dungan JL, Peterson DL (2001) Estimating the foliar biochemical concentration of leaves with reflectance spectrometry. Testing the Kokaly and Clark methodologies, Remote Sensing of Environment, 76, 349-359.

Davalo E, Naim P (1991) Neural Networks, Macmillan, Basingstoke.

Day C (1997) Remote sensing applications which may be addressed by neural networks using parallel processing technology, In Kanellopoulos, I, Wilkinson, G. G., Roli, F., Austin, J. (eds) Neuro-computation in Remote Sensing Data Analysis, Springer, Berlin, pp. 262-279.

Estes J Belward A, Loveland T, Scepan J, Strahler A, Townshend J, Justice C (1999), The way forward, Photogrammetric Engineering and Remote Sensing, 65, 1089-1093.

Fardanesh MT, Ersoy OK (1998) Classification accuracy improvement of neural network classifiers by using unlabeled data, IEEE Transactions on Geoscience and Remote Sensing, 36, 1020-1025.

Fischer MM (1998) Computational neural networks a new paradigm for spatial analysis, Environment and Planning A, 30, 1873-1891.

Fischer MM, Gopal S, Staufer P, Steinnocher K (1997) Evaluation of neural pattern classifiers for a remote sensing application, Geographical Systems, 4, 195-225.

Foody GM (1996) Approaches for the production and evaluation of fuzzy land cover classifications from remotely-sensed data, International Journal of Remote Sensing, 17, 1317-1340.

Foody GM (1999a) Image classification with a neural network: from completely-crisp to fully-fuzzy situations, In Atkinson, P. M. and Tate, N. J. (eds) Advances in Remote Sensing and GIS, Wiley, Chichester, pp. 17-37.

Foody GM (1999b) The significance of border training patterns in classification by a feedforward neural network using backpropagation learning, International Journal of Remote Sensing, 20, 3549-3562.

Foody GM (1999c) The continuum of classification fuzziness in thematic mapping, Photogrammetric Engineering and Remote Sensing, 65, 443-451.

Foody GM (2001) Thematic mapping from remotely sensed data with neural networks: MLP, RBF and PNN based approaches, Journal of Geographical Systems, 3, 217-232.

Foody GM (2004a) Supervised classification by MLP and RBF neural networks with and without an exhaustively defined set of classes, International Journal of Remote Sensing, 25, 3091-3104.

Foody GM (2004b) Thematic map comparison: evaluating the statistical significance of differences in classification accuracy, Photogrammetric Engineering and Remote Sensing, 70, 627-633.

Foody GM, Arora MK (1997) An evaluation of some factors affecting the accuracy of classification by an artificial neural network, International Journal of Remote Sensing, 18, 799-810.

Foody GM, McCulloch MB, Yates WB (1995) Classification of remotely sensed data by an artificial neural network: issues related to training data characteristics. Photogrammetric Engineering and Remote Sensing, 61, 391-401.

Foody GM, Lucas RM, Curran PJ, Honzak M (1997) Non-linear mixture modelling without end-members using an artificial neural network, International Journal of Remote Sensing, 18, 937-953.

Foody GM, Cutler ME, McMorrow J, Pelz D, Tangki H, Boyd DS, Douglas I (2001) Mapping the biomass of Bornean tropical rain forest from remotely sensed data, Global Ecology and Biogeography, 10, 379-387.

Friedl MA, Woodcock C, Gopal S, Muchoney D, Strahler AH, Barker-Schaaf C (2000) A note on procedures used for accuracy assessment in land cover maps derived from AVHRR data, International Journal of Remote Sensing, 21, 1073-1077.

Gamba P, Houshmand B (2001) An efficient neural classification chain of SAR and optical urban images, International Journal of Remote Sensing, 22, 1535-1553.

Gopal S, Woodcock CE, Strahler AH (1999) Fuzzy neural network classification of global land cover from a 1° AVHRR data set, Remote Sensing of Environment, 67, 230-243.

Hall RJ (2000) Applications of remote sensing to forestry – current and future, Forestry Chronicle, 76, 855-857.

Ito Y, Omatu S. (1997) Category classification method using a self-organising neural network, International Journal of Remote Sensing, 18, 829-845.

Ji CY (2000) Land-use classification of remotely sensed data using Kohonen self-organising feature map neural networks, Photogrammetric Engineering and Remote Sensing, 66, 1451-1460.

Jin YQ, Liu C (1997) Biomass retrieval from high-dimensional active/passive remote sensing data by using artificial neural networks, International Journal of Remote Sensing, 18, 971-979.

Kanellopoulos I, Wilkinson GG (1997) Strategies and best practice for neural network image classification, International Journal of Remote Sensing, 18, 711-725.

Kasischke ES, Melack JM. Dobson MC (1997) The use of imaging radars for ecological applications –a review, Remote Sensing of Environment, 59, 141-156.

Kavzoglu T, Mather PM (2003) The use of backpropagating artificial neural networks in land cover classification, International Journal of Remote Sensing, 24, 4907-4938.

Kimes DS, Nelson RF, Fifer ST (2000) Predicting ecologically important vegetation variables from remotely sensed optical/radar data using neuronal networks, In Lek, S. and Guegan, J-F. (eds) Artificial Neuronal Networks: Applications to Ecology and Evolution, Springer, Berlin, pp. 31-44.

Lawrence RL, Ripple WJ (1998) Comparisons among vegetation indices and bandwise regression in a highly disturbed heterogeneous landscape: Mount St. Helens, Washington, Remote Sensing of Environment, 64, 91-102.

Lek S, Giraudel JL, Guegan JF (2000) Neuronal networks: algorithms and architectures for ecologists and evolutionary ecologists, In Lek, S. and Guegan, J-F. (eds) Artificial

Neuronal Networks: Applications to Ecology and Evolution, Springer, Berlin, pp. 3-27.

Lillesand TM, Kiefer RW, Chipman JW (2000) Remote Seining and Image Interpretation, fifth edition, Wiley, New York.

Liu WG, Seto KC, Wu EY, Gopal S, Woodcock CE (2004) ART-MMAP: A neural network approach to subpixel classification, IEEE Transactions on Geoscience and Remote Sensing, 42, 1976-1983.

Loveland TR, Zhu Z, Ohlen DO, Brown JF, Reed BC, Yang L (1999) An analysis of the IGBP global land-cover characterisation process, Photogrammetric Engineering and Remote Sensing, 65, 1021-1032.

Lucas NS, Curran PJ (1999) Forest ecosystem simulation modelling: the role of remote sensing, Progress in Physical Geography, 23, 391-423.

Mather PM (2004) Computer Processing of Remotely-Sensed Images, third edition, Wiley, Chichester.

Pal M, Mather (2004) Assessment of the effectiveness of support vector machines for hyperspectral data, Future Generation Computer Systems,20, 1215-1225.

Paola JD, Schowengerdt RA (1995) A detailed comparison of backpropagation neural network and maximum likelihood classification for urban land use classification, IEEE Transactions on Geoscience and Remote Sensing, 33, 981-996.

Peddle DR, Foody GM, Zhang A, Franklin SE, LeDrew EF (1994) Multisource image classification II: an empirical comparison of evidential reasoning and neural network approaches, Canadian Journal of Remote Sensing, 20, 396-407.

Poth A, Klaus MV, Stein G (2001) Optimisation at multi-spectral land cover classification with fuzzy clustering and the Kohonen feature map, International Journal of Remote Sensing, 22, 1423-1439.

Rollet R, Benie GB, Li W, Wang S, Boucher JM (1998) Image classification algorithm based on the RBF neural network and K-means, International Journal of Remote Sensing, 19, 3003-3009.

Roughgarden J, Running SW, Matson PA (1991) What does remote sensing for ecology? Ecology, 72, 1918-1922.

Scepan J (1999) Thematic validation of high-resolution global land-cover data sets, Photogrammetric Engineering and Remote Sensing, 65, 1051-1060.

Schalkoff R (1992) Pattern Recognition: Statistical, Structural and Neural Approaches Wiley, New York.

Schowengerdt RA (1997) Remote Sensing: Models and Methods for Image Processing, second edition, Academic Press, San Diego.

Shahshahani BM, Landgrebe DA (1994) The effect of unlabeled samples in reducing the small sample-size problem and mitigating the Hughes phenomenon, IEEE Transactions on Geoscience and Remote Sensing, 32, 1087-1095.

Smith JA (1993) LAI inversion using a back-propagating neural network trained with a multiple scattering model, IEEE Transactions on Geoscience and Remote Sensing, 31, 1102-1106.

Staufer P, Fischer MM (1997) Spectral pattern recognition by a two-layer perceptron: effects of training set size, In: Kanellopoulos, I., Wilkinson GG , Roli F, Austin J (eds) Neuro-computation in Remote Sensing Data Analysis, Springer, Berlin, pp. 105-116.

Strahler AH (1980) The use of prior probabilities in maximum likelihood classification of remotely sensed data, Remote Sensing of Environment, 10, 135-163.

Specht DF (1990) Probabilistic neural networks, Neural Networks, 3, 109-118.

Tatem AJ, Lewis HG., Atkinson PM, Nixon MS (2001) Super-resolution target identification from remotely sensed images using a Hopfield neural network, IEEE Transactions on Geoscience and Remote Sensing, 39, 781-796.

Tatem AJ, Lewis HG, Atkinson PM, Nixon MS (2003) Increasing the spatial resolution of agricultural land cover maps using a Hopfield neural network, International Journal of Geographical Information Science,17, 647-672.

Townshend JRG (1992) Land cover, International Journal of Remote Sensing, 13, 1319-1328.

Index

A

abundance 56-63
adaptive agents
109-124
-, individual based 112
-, state variable-based 114
artificial neural networks
49-66, 151-165, 169-185, 187-234, 239-250,255-270, 275-291, 293-304, 309-322, 325-344, 369-382, 431-442, 445-456, 459-473
-, non-supervised 49-66, 172-174, 190-206, 221-232, 277-279, 325-344
-, probabilistic 445-456
-, recurrent 207-218, 255-270, 325-344
-, sensitivity analysis 239-250, 255-270, 293-304, 331-343
-, supervised 151-165, 175-180, 207-220, 239-250, 255-270, 275-291, 293-304, 309-322, 325-344, 369-382, 431-442, 445-456, 459-473

B

bootstrap data re-sampling 385-406
back-propagation 151-165, 175-180, 207-220, 239-250, 255-270, 275-291, 293-304, 309-322, 325-344, 369-382, 431-442, 445-456, 459-473

C

chlorophyll *a* 309-322, 409-427
cellular automata 125-146
-, self-replicating125-146
classification trees 151-165
clustering 49-66, 187-234325-344, 431-442
crossover 73, 91, 349-350

D

distance matrix map 329-344
diversity 151-156, 169-185, 187-234, 239-250
DNA 125

E

evolutionary algorithms 85-102, 116-117, 347-366
-, hybrid 347-366
-, structure optimization 351-355
-, parameter optimization 356-357

F

fish 293-304, 385-406
-, age estimation 445-456
-, freshwater 293-304
-, marine 385-406, 445-456
-, stock recruitment 385-406
fitness 352
forecasting 78-79, 93-96, 109-123, 149-165, 169-185, 187-234, 255-270, 275-291, 309-322, 325-344, 347-366, 369-382, 409-427
-, time-series 85-102, 109-123, 187-234, 255-270, 275-291, 325-344, 347-366, 369-382, 409-427
-, steady state 3-12, 151-165, 309-322, 169-185, 187-234, 309-322, 385-406
fuzzy logic 3-12, 385-406

G

genetic algorithms 69-84, 351-357
Germany169-186

H

hardware design 125-146

hidden nodes 151-165, 175-180, 207-220, 239-250, 255-270, 275-291, 293-304, 309-322, 325-344, 369-382, 431-442, 445-456, 459-473
hybrid evolutionary algorithm 347-366

I
individual based adaptive agents 112

J
Japan 109-123, 325-344, 347-366, 409-427

K
K-means 329-344
Korea 187-235, 255-270, 325-344, 347-366

L
lakes 109-123, 309-322, 325-344, 347-366, 409-427
-, Emir 309-322
-, Kasumigaura 109-123, 325-344, 347-366, 409-427
-, Keban 309-322
-, Morgan 309-322
-, Soyang 325-344, 347-366

M
macroinvertebrates 151-156, 169-185, 187-234, 239-250
Murray 275-291
monsoon 325-344, 347-366
mutation 73, 91, 349-350

N
Nakdong 255-270
Netherlands 151-167
nodes 151-165, 175-180, 207-220, 239-250, 255-270, 275-291, 293-304, 309-322, 325-344, 369-382, 431-442, 445-456, 459-473

O
ordinary differential equations 94-95, 114-121, 409-427
ordination 49-66, 187-234325-344, 431-442

P
partitioning map 329-344
pattern recognition 431-442, 445-456, 459-473
phytoplankton 93-96, 114-122, 255-270, 282-290, 309-322, 325-344, 347-366, 309-427, 431-442
-, chlorophyll *a* 309-322, 409-427
-, identification 431-442
-, population dynamics 93-96, 109-123, 255-270, 275-291, 325-344, 347-366
-, species 93-96, 109-123, 255-270, 275-291, 325-344, 347-366, 431-442

Q
qualitative reasoning 15-44

R
recurrent artificial neural networks 207-218, 255-270, 325-344
reproduction 73, 91, 349-350
rivers 255-270, 275-291, 293-304
-, Murray 275-291
-, Nakdong 255-270

rule sets 9-11, 95-97, 117, 347-366, 390-392
-, evolved 95-97, 117, 347-366
-, fuzzy 9-11, 390-392

S
sensitivity analysis 239-250, 255-270, 293-304, 331-343
Soyang 325-344, 347-366
streams 151-165, 169-185, 187-234, 239-250

T
training 151-165, 175-180, 207-220, 239-250, 255-270, 275-291, 293-304, 309-322, 325-344, 369-382, 431-442, 445-456, 459-473
time lag 255-270, 325-344
Turkey 309-322

U
U-matrix 60-63, 329-344

V
validation 151-165, 175-180, 207-220, 239-250, 255-270, 275-291, 293-304, 309-322, 325-344, 369-382, 431-442
Von Neumann 128-131

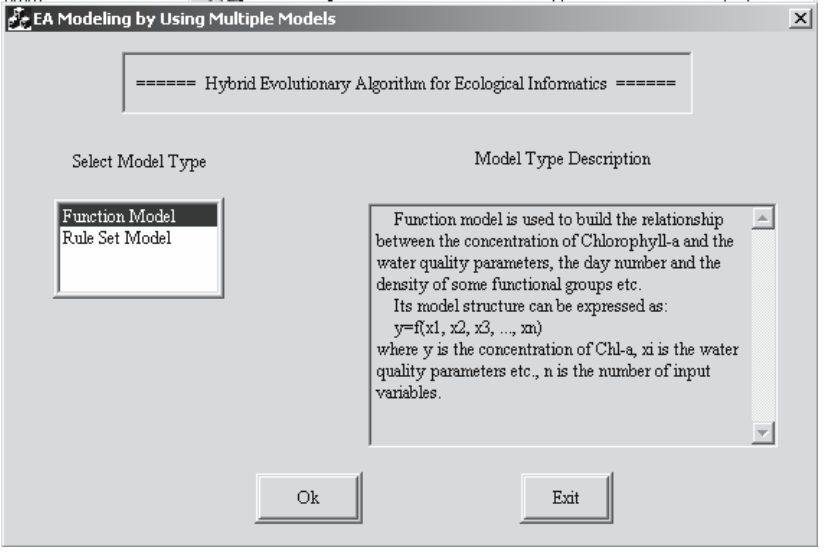
W
weights 151-165, 175-180, 207-220, 239-250, 255-270, 275-291, 293-304, 309-322, 325-344, 369-382, 431-442, 445-456, 459-473

Z
zooplankton 369-382

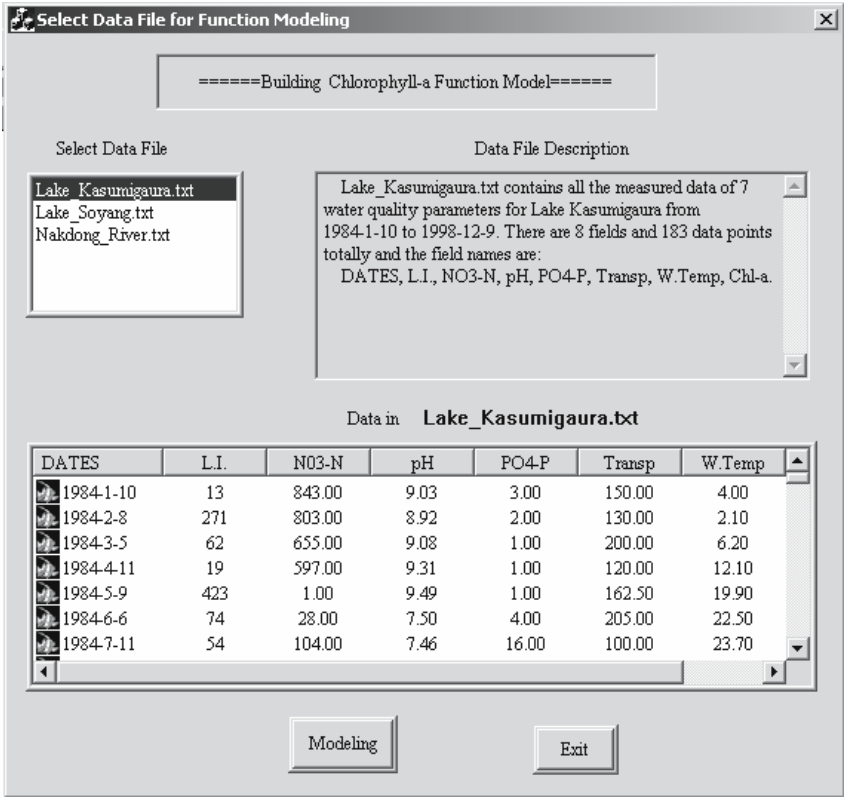
Appendix to Chapter 17
Instructions for Demo Version of the Hybrid Evolutionary Algorithm on Attached CD

This software is used to demonstrate how to automatically create either a function and rule set for ecological data by using the hybrid evolutionary algorithm (HEA).

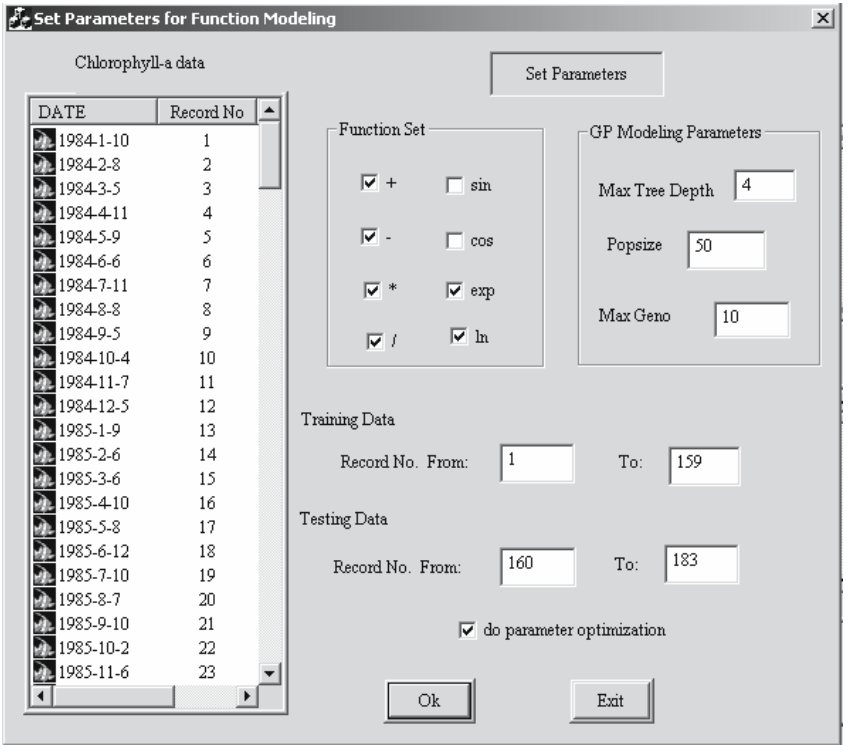
You can use the Demo Version by the following steps:
1. Click the exe file HEAdemo.exe and the following screen will appear.


EA Modeling by Using Multiple Models
====== Hybrid Evolutionary Algorithm for Ecological Informatics ======
Select Model Type
Function Model
Rule Set Model
Model Type Description
Function model is used to build the relationship between the concentration of Chlorophyll-a and the water quality parameters, the day number and the density of some functional groups etc.
Its model structure can be expressed as:
y=f(x1, x2, x3, ..., xn)
where y is the concentration of Chl-a, xi is the water quality parameters etc., n is the number of input variables.
Ok
Exit

2. Double click the “Function Model” to choose the model type. Click “Ok” to start the function modelling and the following screen will appear.


Select Data File for Function Modeling
======Building Chlorophyll-a Function Model======
Select Data File
Lake_Kasumigaura.txt
Lake_Soyang.txt
Nakdong_River.txt
Data File Description
Lake_Kasumigaura.txt contains all the measured data of 7 water quality parameters for Lake Kasumigaura from 1984-1-10 to 1998-12-9. There are 8 fields and 183 data points totally and the field names are:
DATES, L.I., NO3-N, pH, PO4-P, Transp, W.Temp, Chl-a.
Data in Lake_Kasumigaura.txt
DATES L.I. N03-N pH PO4-P Transp W.Temp
1984-1-10 13 843.00 9.03 3.00 150.00 4.00
1984-2-8 271 803.00 8.92 2.00 130.00 2.10
1984-3-5 62 655.00 9.08 1.00 200.00 6.20
1984-4-11 19 597.00 9.31 1.00 120.00 12.10
1984-5-9 423 1.00 9.49 1.00 162.50 19.90
1984-6-6 74 28.00 7.50 4.00 205.00 22.50
1984-7-11 54 104.00 7.46 16.00 100.00 23.70
Modeling
Exit

3. There are three data sets which are preset and fixed to test the efficiency of HEA in function modelling. You can choose any of them by double clicking the file name. Suppose we select the data file “Lake Kasumigaura.txt” and then click “Modeling”. The following screen will appear.



4. This window is used to complete the parameter settings of HEA and modelling experiment. The simplest way is to use the default settings and click "Ok" to go to the next step. However as those default values are set only for the purpose of quick running and usually can not make sure to achieve good result, you had better change some parameters by yourself if you do not mind the running time and hope to get better result. The meaning of those parameters are listed below:

Function Set:

There are 4 arithmetic operators ("+","-","*","/") and 4 functions ("sin", "cos", "exp", "ln"). HEA will use these predefined operators and functions to construct the model expression. You can click the corresponding box alternatively to choose it or not.

GP Modeling Parameters:

"Max Tree Depth": the maximal tree depth of the function model. The larger this value is, the more complicated is the model. It ranges from 3 to 10.

"**Popsize**": the size of the initial population. It ranges from 50 to 1000.

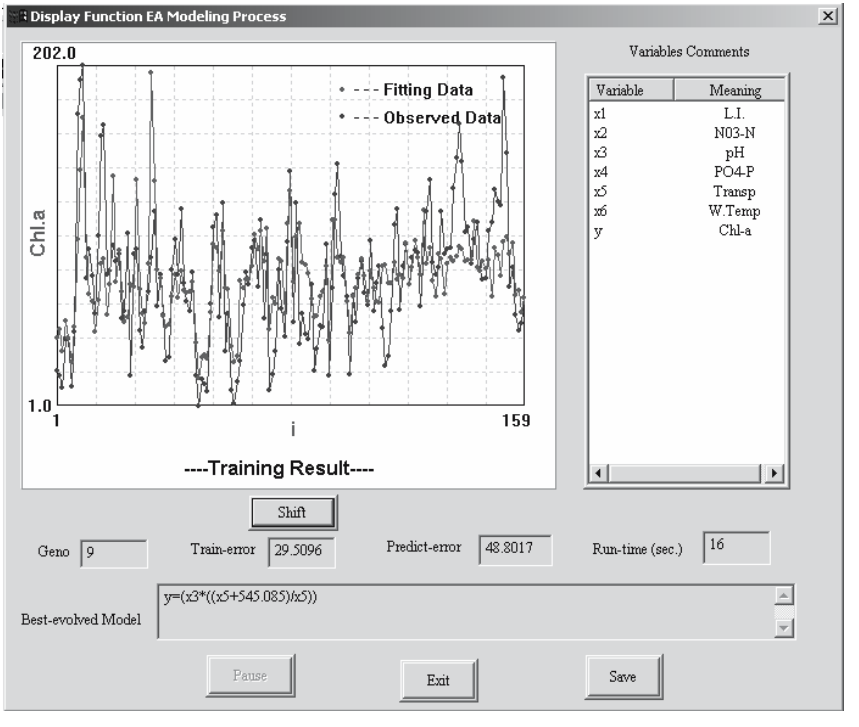
"**Max Geno**": the maximal number of generations in one run. It defines the termination criterion. It ranges from 5 to 500.

"Training data":

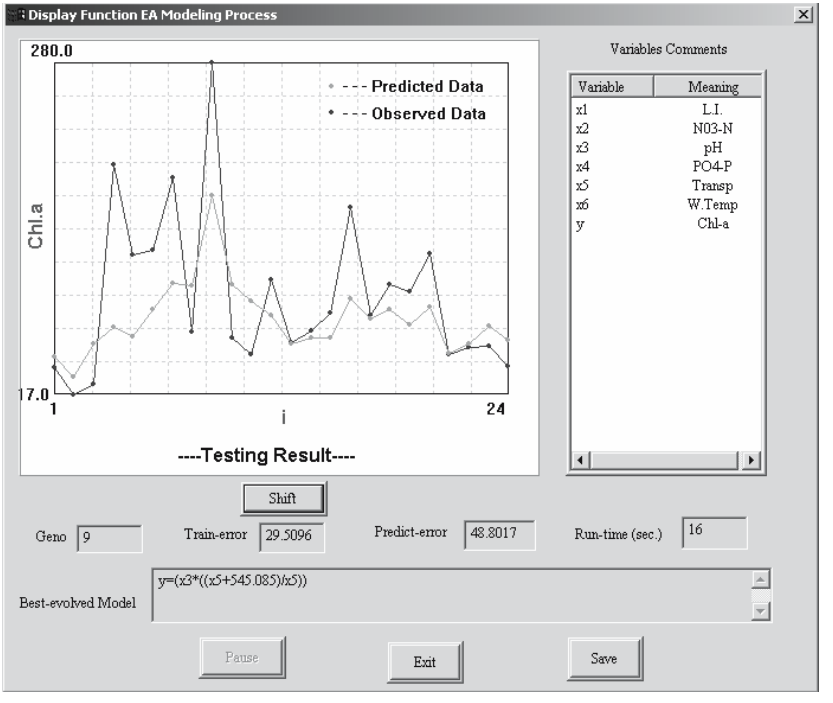
You need to define the starting record no. and the ending record no. of the training data by referring to the "Date" and the "Record no" in the left table. The training data can start from any year but must be continuous.

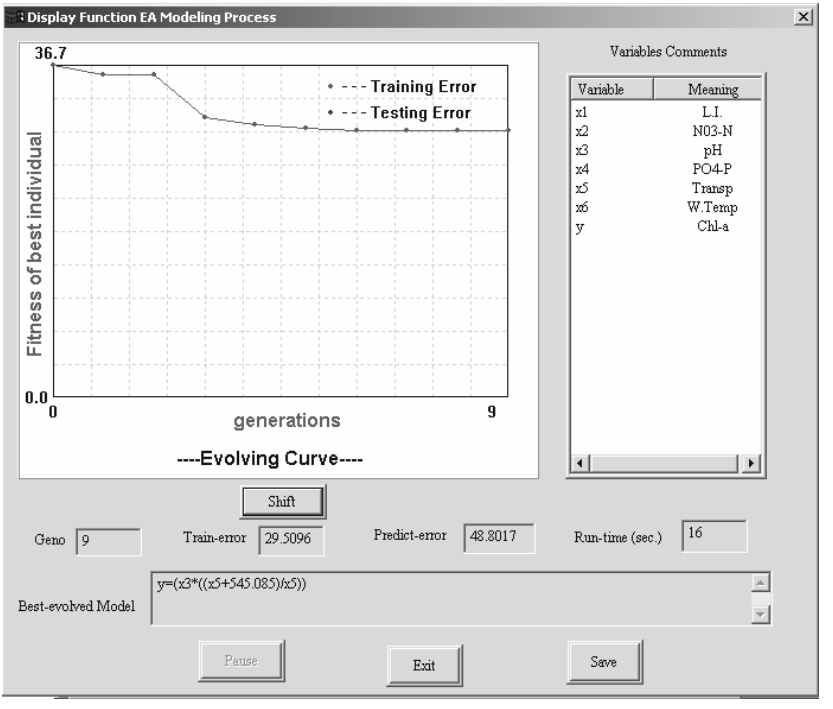
"Testing data":
Similarly you need to define the starting record no. and the ending record no. of the testing data by referring to the "Date" and the "Record no" in the left table. The testing data can also start from any year but must be continuous. In the default settings, we usually fix the training years and testing years as the same as we have done in the paper. In addition in order to reduce running time, in this demo version we only use observed data rather than interpolated data to do the modelling.
"do parameter optimisation": It means in every generation we will optimise all the parameters in each model by using a general GA as described in the paper. You can choose to perform this process or not by click the box alternatively. Usually the result will get better compared with not using this process.

Once you complete all the parameter settings, click "Ok" to start the modelling procedure. Then the following screen will appear.

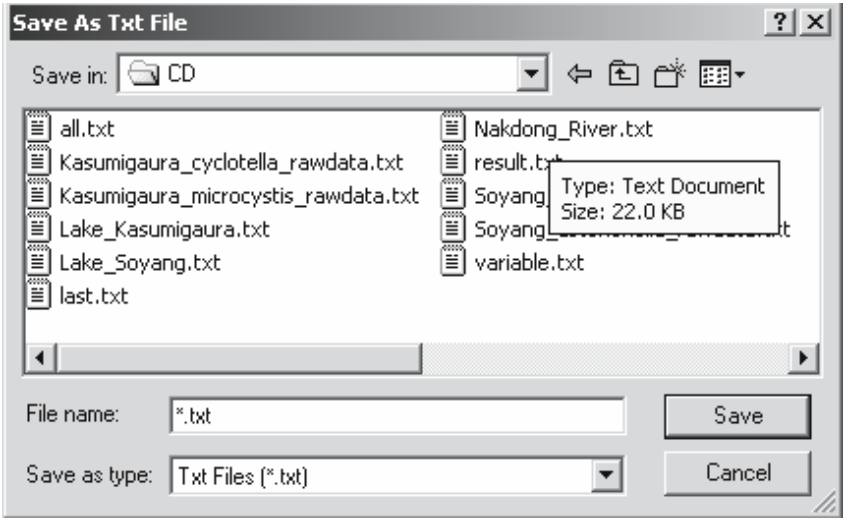


5. This window will display the real-time modelling procedure. The information shown here includes the graphs for fitting training data and predicting the testing data and the evolving curve generation by generation; the meanings of the input variables and output variable; the current generation; the training error and the testing error, running time; and the expression of the best-evolved model. You can click "shift" to view different graphs as follows.

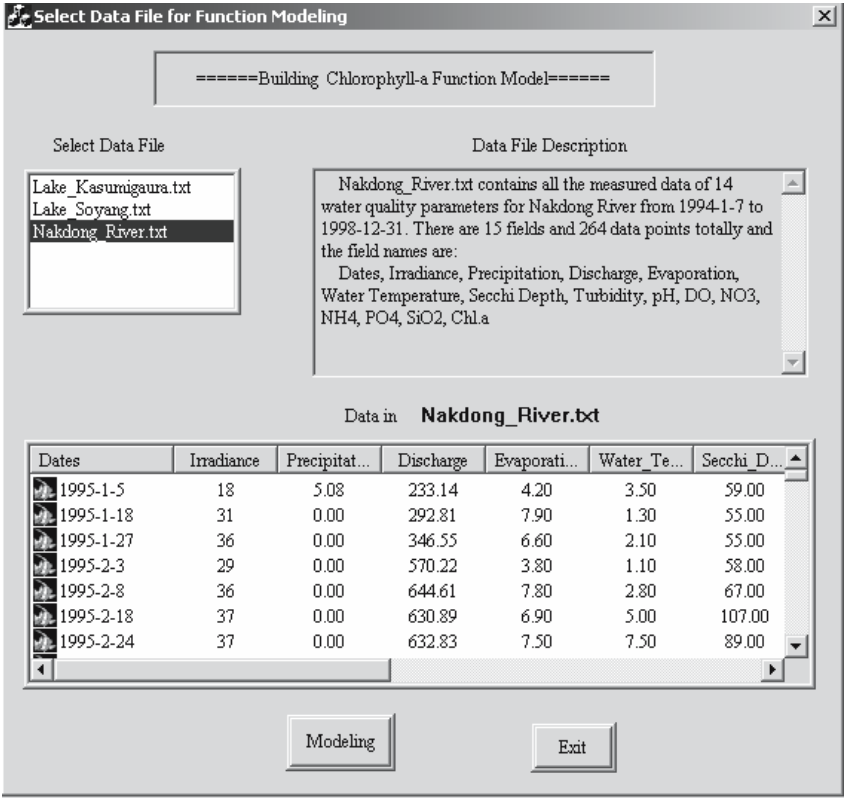

Display Function EA Modeling Process
280.0
--- Predicted Data
--- Observed Data
Chl.a
17.0
1
i
24
----Testing Result----
Variables Comments
Variable
Meaning
x1 L.I.
x2 N03-N
x3 pH
x4 PO4-P
x5 Transp
x6 W.Temp
y Chl-a
Shift
Geno 9
Train-error 29.5096
Predict-error 48.8017
Run-time (sec.) 16
Best-evolved Model
y=(x3*((x5+545.085)/x5))
Pause
Exit
Save


Display Function EA Modeling Process
36.7
--- Training Error
--- Testing Error
Fitness of best individual
0.0
0
generations
9
----Evolving Curve----
Variables Comments
Variable
Meaning
x1 L.I.
x2 N03-N
x3 pH
x4 PO4-P
x5 Transp
x6 W.Temp
y Chl-a
Shift
Geno 9
Train-error 29.5096
Predict-error 48.8017
Run-time (sec.) 16
Best-evolved Model
y=(x3*((x5+545.085)/x5))
Pause
Exit
Save

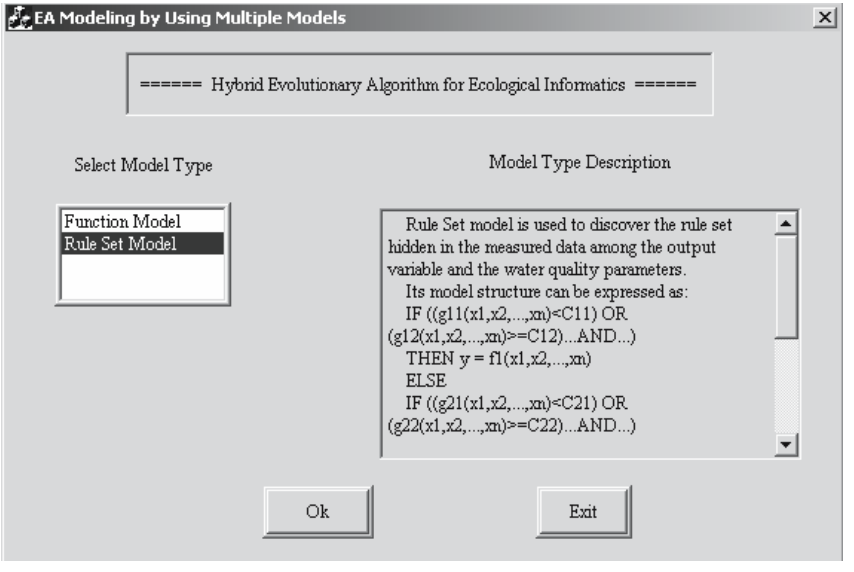
6. When the predefined “Max Geno” is reached, the HEA will stop running and you can click “Save” to save the modelling results as a text file in the selected folder.



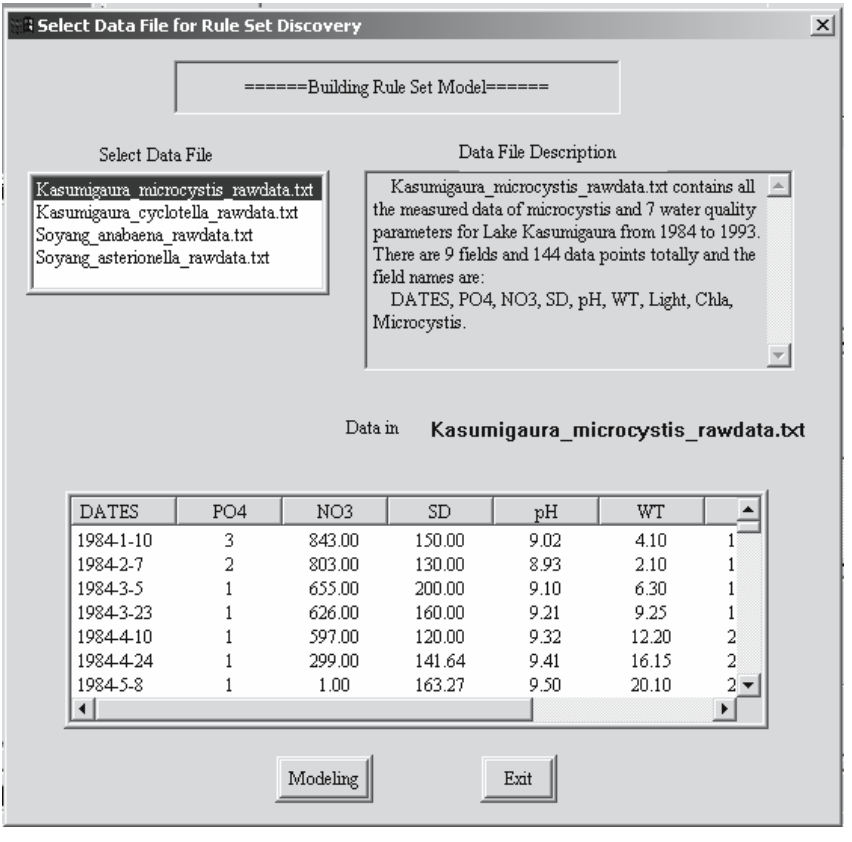
7.Click “Exit” to return to the following window so that you can select another data file to do the function modelling by following the previous Step3 ~ Step6.


Select Data File for Function Modeling
======Building Chlorophyll-a Function Model======
Select Data File
Lake_Kasumigaura.txt
Lake_Soyang.txt
Nakdong_River.txt
Data File Description
Nakdong_River.txt contains all the measured data of 14 water quality parameters for Nakdong River from 1994-1-7 to 1998-12-31. There are 15 fields and 264 data points totally and the field names are:
Dates, Irradiance, Precipitation, Discharge, Evaporation, Water Temperature, Secchi Depth, Turbidity, pH, DO, NO3, NH4, PO4, SiO2, Chl.a
Data in Nakdong_River.txt
Dates Irradiance Precipitat... Discharge Evaporati... Water_Te... Secchi_D...
1995-1-5 18 5.08 233.14 4.20 3.50 59.00
1995-1-18 31 0.00 292.81 7.90 1.30 55.00
1995-1-27 36 0.00 346.55 6.60 2.10 55.00
1995-2-3 29 0.00 570.22 3.80 1.10 58.00
1995-2-8 36 0.00 644.61 7.80 2.80 67.00
1995-2-18 37 0.00 630.89 6.90 5.00 107.00
1995-2-24 37 0.00 632.83 7.50 7.50 89.00
Modeling
Exit

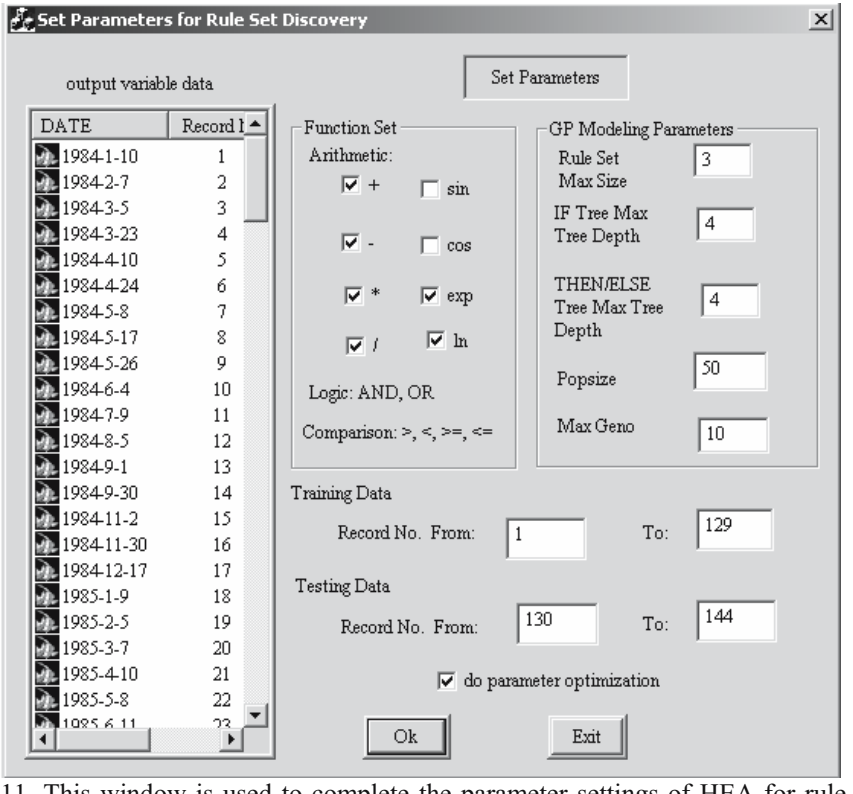
8. If you want to try “rule set model”, click “exit” again and return to the following initial screen.


EA Modeling by Using Multiple Models
====== Hybrid Evolutionary Algorithm for Ecological Informatics ======
Select Model Type
Function Model
Rule Set Model
Model Type Description
Rule Set model is used to discover the rule set hidden in the measured data among the output variable and the water quality parameters.
Its model structure can be expressed as:
IF ((g11(x1,x2,...,xn)<C11) OR (g12(x1,x2,...,xn)>=C12)...AND...)
THEN y = f1(x1,x2,...,xn)
ELSE
IF ((g21(x1,x2,...,xn)<C21) OR (g22(x1,x2,...,xn)>=C22)...AND...)
Ok
Exit

9. Double click the "Rule Set Model" and click "Ok" to start the rule set modelling and the following screen will appear.


Select Data File for Rule Set Discovery
======Building Rule Set Model======
Select Data File
Kasumigaura_microcystis_rawdata.txt
Kasumigaura_cyclotella_rawdata.txt
Soyang_anabaena_rawdata.txt
Soyang_asterionella_rawdata.txt
Data File Description
Kasumigaura_microcystis_rawdata.txt contains all the measured data of microcystis and 7 water quality parameters for Lake Kasumigaura from 1984 to 1993. There are 9 fields and 144 data points totally and the field names are:
DATES, PO4, NO3, SD, pH, WT, Light, Chla, Microcystis.
Data in Kasumigaura_microcystis_rawdata.txt
DATES PO4 NO3 SD pH WT
1984-1-10 3 843.00 150.00 9.02 4.10 1
1984-2-7 2 803.00 130.00 8.93 2.10 1
1984-3-5 1 655.00 200.00 9.10 6.30 1
1984-3-23 1 626.00 160.00 9.21 9.25 1
1984-4-10 1 597.00 120.00 9.32 12.20 2
1984-4-24 1 299.00 141.64 9.41 16.15 2
1984-5-8 1 1.00 163.27 9.50 20.10 2
Modeling
Exit

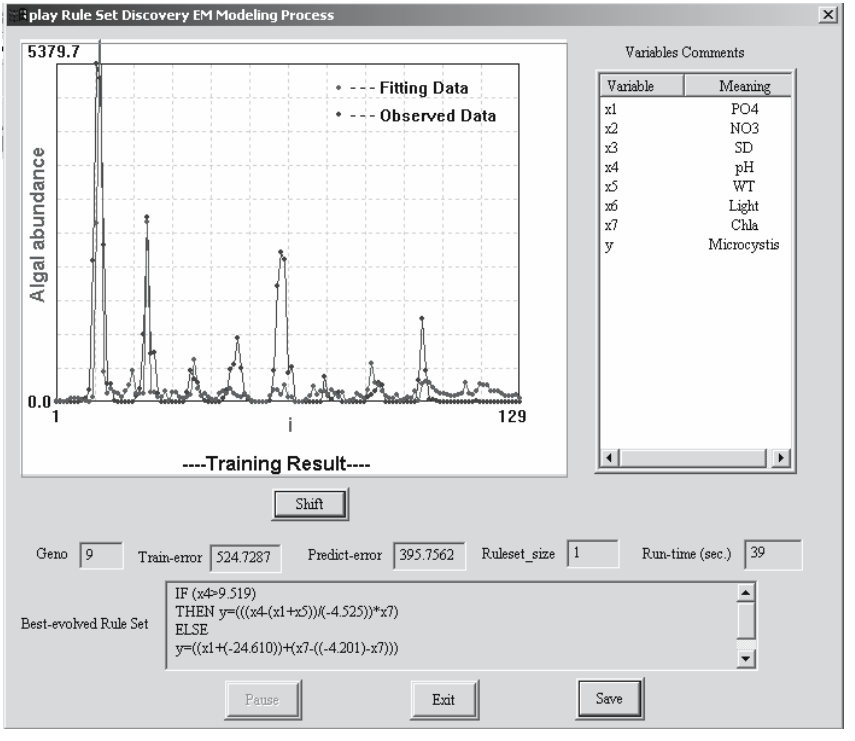
10. There are four data sets which are preset and fixed to test the efficiency of HEA in rule set modelling as we have used in the paper. You can choose any of them by double clicking the file name. Suppose we select the data file "Kasumigaura_Microcystis_rawdata.txt" and then click "Modeling". The following screen will appear.



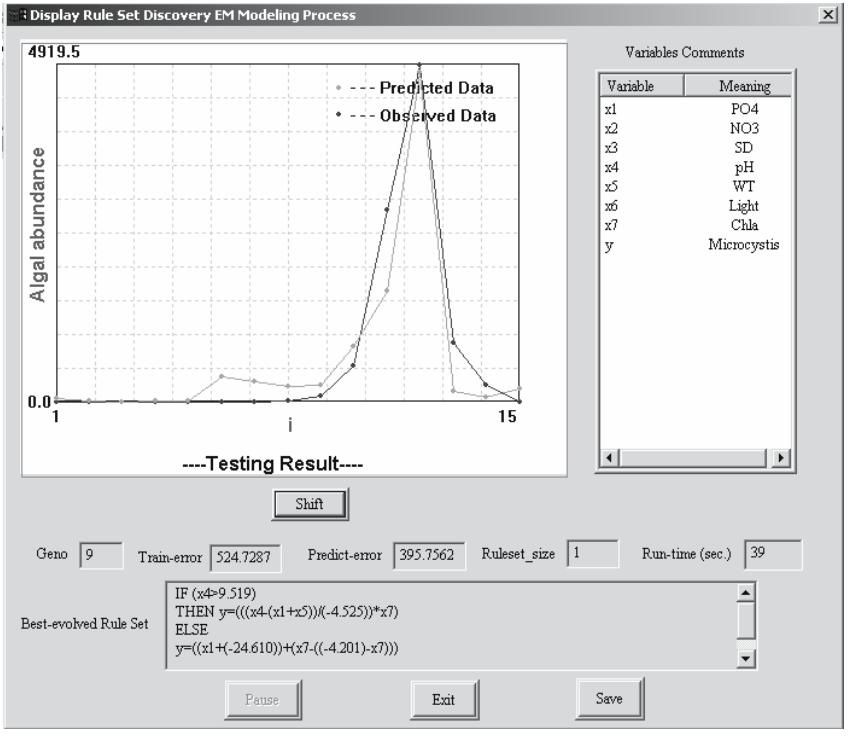
11. This window is used to complete the parameter settings of HEA for rule set modeling. Most parameters have the same meanings as described in Step 4. We only mention some differences here.

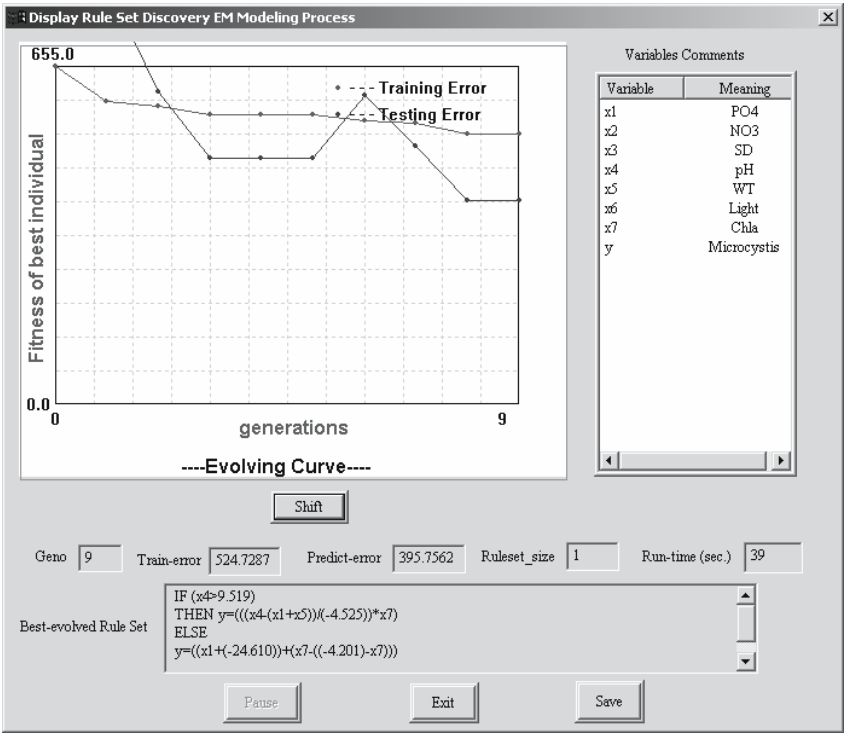
As for the **"Function Set**", it consists of three function sets: Arithmetic, Logic and comparison. They are used to construct the IF Tree and THEN/ELSE Tree in the rule set. You can choose the arithmetic function set by clicking the corresponding box but for the other two function sets, they are always selected by default.
"Rule Set Max Size": the maximal size of a rule set. The size of a rule set means the number of the IF-branches contained in the rule set . Obviously the larger this value is, the more complicated is the rule set. It ranges from 1 to 5.

When you complete all the parameter settings, click "Ok" to start the modelling procedure. The following screen will appear.

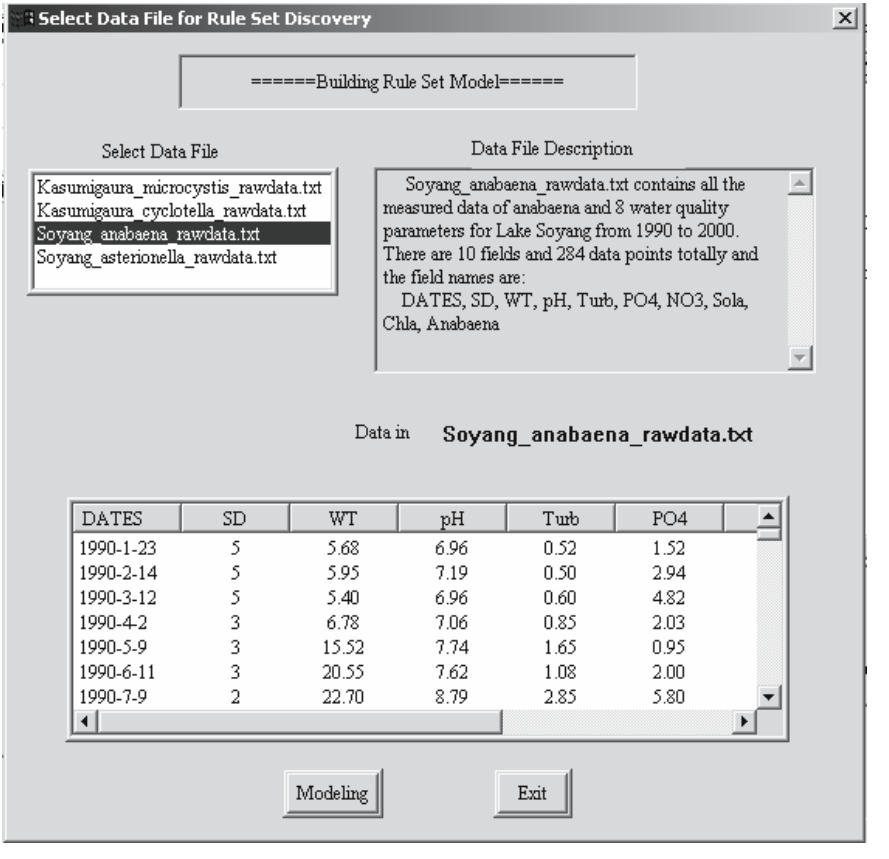


12. This window will display the modelling procedure of rule set dynamically. You can click “Shift” to view different graphs as follows.


Display Rule Set Discovery EM Modeling Process
4919.5
--- Predicted Data
--- Observed Data
Algal abundance
0.0
1
i
15
----Testing Result----
Variables Comments
Variable
Meaning
x1 PO4
x2 NO3
x3 SD
x4 pH
x5 WT
x6 Light
x7 Chla
y Microcystis
Shift
Geno 9
Train-error 524.7287
Predict-error 395.7562
Ruleset_size 1
Run-time (sec.) 39
Best-evolved Rule Set
IF (x4>9.519)
THEN y=(((x4-(x1+x5))/(-4.525))*x7)
ELSE
y=((x1+(-24.610))+(x7-((-4.201)-x7)))
Pause
Exit
Save


Display Rule Set Discovery EM Modeling Process
655.0
0.0
Fitness of best individual
0
9
generations
Training Error
Testing Error
----Evolving Curve----
Variables Comments
Variable
Meaning
x1 PO4
x2 NO3
x3 SD
x4 pH
x5 WT
x6 Light
x7 Chla
y Microcystis
Shift
Geno 9
Train-error 524.7287
Predict-error 395.7562
Ruleset_size 1
Run-time (sec.) 39
Best-evolved Rule Set
IF (x4>9.519)
THEN y=(((x4-(x1+x5))/(-4.525))*x7)
ELSE
y=((x1+(-24.610))+(x7-((-4.201)-x7)))
Pause
Exit
Save

13.When the modelling procedure is finished you can save the result and click "Exit" to go back to the following window to choose a new data set.


Select Data File for Rule Set Discovery
======Building Rule Set Model======
Select Data File
Kasumigaura_microcystis_rawdata.txt
Kasumigaura_cyclotella_rawdata.txt
Soyang_anabaena_rawdata.txt
Soyang_asterionella_rawdata.txt
Data File Description
Soyang_anabaena_rawdata.txt contains all the measured data of anabaena and 8 water quality parameters for Lake Soyang from 1990 to 2000. There are 10 fields and 284 data points totally and the field names are:
DATES, SD, WT, pH, Turb, PO4, NO3, Sola, Chla, Anabaena
Data in Soyang_anabaena_rawdata.txt
DATES SD WT pH Turb PO4
1990-1-23 5 5.68 6.96 0.52 1.52
1990-2-14 5 5.95 7.19 0.50 2.94
1990-3-12 5 5.40 6.96 0.60 4.82
1990-4-2 3 6.78 7.06 0.85 2.03
1990-5-9 3 15.52 7.74 1.65 0.95
1990-6-11 3 20.55 7.62 1.08 2.00
1990-7-9 2 22.70 8.79 2.85 5.80
Modeling
Exit